

Pricing in Networks

Citation for published version (APA):

van der Kraaij, A. F. (2004). *Pricing in Networks*. [Doctoral Thesis, Maastricht University]. Datawyse / Universitaire Pers Maastricht. <https://doi.org/10.26481/dis.20041112ak>

Document status and date:

Published: 01/01/2004

DOI:

[10.26481/dis.20041112ak](https://doi.org/10.26481/dis.20041112ak)

Document Version:

Publisher's PDF, also known as Version of record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.umlib.nl/taverne-license

Take down policy

If you believe that this document breaches copyright please contact us at:

repository@maastrichtuniversity.nl

providing details and we will investigate your claim.

PRICING IN NETWORKS

Proefschrift

ter verkrijging van de graad van doctor
aan de Universiteit Maastricht,
op gezag van de Rector Magnificus,
Prof. mr. G.P.M.F. Mols,
volgens het besluit van het College van Decanen,
in het openbaar te verdedigen
op vrijdag 12 november 2004 om 12.00 uur

door

Anton Frederik van der Kraaij

Promotor:

Prof. dr. ir. C.P.M. van Hoesel

Beoordelingscommissie:

Prof. dr. ir. A. Kolen (voorzitter)

Prof. dr. M. Labbé (Université Libre de Bruxelles)

Prof. dr. R. Müller

dr. D. Romero Morales (University of Oxford)

Pricing in Networks

©2004, Anton F. van der Kraaij

Proefschrift Universiteit Maastricht

ISBN 90-9018604-2

Omslagillustratie: ©2004, Eisjen van der Kraaij-Schaaf. Alle rechten voorbehouden.

Dit onderzoek werd financieel mede mogelijk gemaakt door France Télécom Research & Development.

Contents

Introduction	To Marguérita	175
Chapter I		1
Chapter II		25
Chapter III		45
Chapter IV		65
Chapter V		85
Chapter VI		105
Chapter VII		125
Chapter VIII		145
Chapter IX		165
Chapter X		185
Chapter XI		205
Chapter XII		225
Chapter XIII		245
Chapter XIV		265
Chapter XV		285
Chapter XVI		305
Chapter XVII		325
Chapter XVIII		345
Chapter XIX		365
Chapter XX		385
Chapter XXI		405
Chapter XXII		425
Chapter XXIII		445
Chapter XXIV		465
Chapter XXV		485
Chapter XXVI		505
Chapter XXVII		525
Chapter XXVIII		545
Chapter XXIX		565
Chapter XXX		585
Chapter XXXI		605
Chapter XXXII		625
Chapter XXXIII		645
Chapter XXXIV		665
Chapter XXXV		685
Chapter XXXVI		705
Chapter XXXVII		725
Chapter XXXVIII		745
Chapter XXXIX		765
Chapter XL		785
Chapter XLI		805
Chapter XLII		825
Chapter XLIII		845
Chapter XLIV		865
Chapter XLV		885
Chapter XLVI		905
Chapter XLVII		925
Chapter XLVIII		945
Chapter XLIX		965
Chapter L		985

Contents

Acknowledgements	15
1 Introduction	17
1.1 A Stackelberg Pricing Game	17
1.2 Applications	20
1.2.1 Transportation Industry	20
1.2.2 Telecommunications Industry	21
1.3 Variations on the NTP considered	23
1.3.1 Pricing Strategy	23
1.3.2 Network Structure	24
1.3.3 All-service NTP	25
1.4 Related Tarification Problems	26
1.5 Outline of the Thesis	27
2 The Network Tarification Problem	29
2.1 A Special Case of Bilevel Programming	29
2.2 Pricing Strategies	32
2.3 Model	34
2.4 Investigating the Leader's Revenue	35
2.5 The Shortest Path Graph Model	39
2.5.1 Remodeling the Network	39
2.5.2 Reduction Methods	42
2.5.3 River Tarification Problem	47
2.5.4 Insights into the Shortest Path Graph Model	47
3 Complexity	53
3.1 NP-completeness	54
3.1.1 Network Tarification Problem	54
3.1.2 River Tarification Problem	57
3.1.3 All-service NTP	59
3.2 Polynomially Solvable Special Cases	59
3.2.1 Bounded Arcs Linear Tarification Problem	60
3.2.2 Parametric Tarification Problem	64
3.2.3 Fixed Path Linear Tarification Problem	66
3.2.4 Multi-Commodity Fixed Path Tarification Problem	66
3.2.5 Single Arc Linear Tarification Problem	68

3.2.6	Single Arc Fixed Charge Tarification Problem	69
3.2.7	Bounded Number of Commodities for RTP	71
3.2.8	Special Cost Structure for RTP	71
4	Approximation	73
4.1	River Tarification Problem	73
4.1.1	$ T $ -approximation Algorithm	74
4.1.2	$O(\log(T))$ -approximation Algorithm for Bounded Input Data	75
4.1.3	Tightness of the Approximation	77
4.1.4	Numerical Results	79
4.2	Inapproximability of the All-service NTP	80
4.3	Failing Approximation Algorithms for the NTP	82
5	Optimizing a Linear Pricing Strategy	85
5.1	Problem Formulations	85
5.1.1	Arc Formulation	86
5.1.2	Path Formulations	88
5.1.3	Arc-Path Formulation	90
5.2	Branch and Bound	91
5.2.1	Branching Rules	92
5.2.2	Node Processing	92
5.3	Numerical Results	95
5.4	Tables	99
6	Optimizing All Pricing Strategies	105
6.1	Model	105
6.2	Problem Formulations	107
6.2.1	Arc Formulation	107
6.2.2	Path Formulation	108
6.3	Branch and Bound	110
6.3.1	Branching Rules	110
6.3.2	Node Processing	110
6.4	Results	111
6.4.1	Running Times	111
6.4.2	The Best Pricing Strategy	112
6.4.3	Consequences for the Clients	113
6.4.4	Conclusion	114
6.5	Tables	115
7	Summary and Concluding Remarks	119
	Bibliography	121
	Index	127
	Samenvatting	131
	About the Author	133

List of Tables

2.1	SPGM statistics for unit grid graphs.	50
4.1	Quality of uniform tarification on France Télécom instances.	79
5.1	Description of France Télécom instances.	99
5.2	SPGM statistics for France Télécom instances.	99
5.3	SPGM statistics for random instances.	100
5.4	CPU times for France Télécom instances.	101
5.5	CPU times for networks with 100 nodes and 2000 arcs.	102
5.6	CPU times for networks with 75 nodes and 4000 arcs.	103
6.1	Description of France Télécom instances.	115
6.2	SPGM generation and path number statistics.	115
6.3	CPU times for path MIP formulations.	116
6.4	CPU times for branch and bound algorithms.	117
6.5	Revenue per pricing strategy.	118
6.6	Number and percentage of clients served.	118
6.7	Average revenue per client (indexed).	118

List of Figures

1.1	The NTP with two clients and a single tariff arc.	18
1.2	The NTP with a single client and two tariff arcs.	19
1.3	A simple freight tariff setting problem.	20
1.4	Interconnections market.	22
1.5	Applications of the network tarification problem.	22
1.6	Several examples of pricing strategies.	24
1.7	Worst case scenario for the ALL-SERVICE NTP.	25
2.1	Upper bound on revenue is not reached.	36
2.2	Optimal tariffs may be negative.	37
2.3	Upper bound on revenue $\mathcal{O}(\log(T))$ far from the optimum.	38
2.4	Network for Example 2.3.	40
2.5	Shape of the leader's objective function.	40
2.6	Shortest Path Graph Model for $ T = 3$	41
2.7	Detailed view of tariff arc.	43
2.8	A small network with $ T = 3$ (Example 2.5).	45
2.9	Shortest Path Graph Model for Example 2.5.	45
2.10	Dominated paths example.	46
2.11	Shortest Path Graph Model for RTP.	48
2.12	Network with euclidean distances.	49
2.13	Network for Example 2.7, for $ T = 3$	49
3.1	Subnetwork for $(\ell_{i1} \vee \ell_{i2} \vee \ell_{i3})$	54
3.2	Network for $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_4 \vee \bar{x}_2)$	56
3.3	Subnetwork for variable x_i , $i \in \{1, \dots, n\}$	56
3.4	Network for $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_4)$	58
3.5	Increasing the tariff on the shortest path tree for PTP.	65
3.6	A single arc fixed charge tarification.	69
3.7	Rotating the fixed charge tariff function.	70
3.8	Special cost structure of network ($ K = 2, T = 3$).	72
4.1	Optimal fractional tariffs exist for the NTP.	74
4.2	Staircase function $f(x)$ with inscribed rectangle.	75
4.3	Illustration for the proof of Lemma 4.2.	76
4.4	The analysis of uniform tarification policies is tight.	78

4.5	Reduction of INDEPENDENT SET to ALL-SERVICE RTP.	80
4.6	Worst-case network for k -tariff arcs algorithm ($ T = 5, k \leq T $). . . .	83
4.7	Worst-case network for null-path algorithm.	83
5.1	Branching rules for branch and bound algorithm.	93
6.1	Pricing strategies.	106

Notations

The notations used throughout this thesis are defined in the text where they appear the first time. Here, we give a list of notations that are often used, for easy reference.

N	Set of nodes.	34
A	Set of arcs.	34
T	Set of tariff arcs.	34
F	Set of fixed cost arcs.	34
G	Directed graph with nodes N and arcs A	34
K	Set of clients.	34
d_k	Demand of client $k \in K$	34
s_k	Source node of demand for client $k \in K$	34
t_k	Target node of demand for client $k \in K$	34
P_k	Set of all possible paths from s_k to t_k for $k \in K$	34
T_p	Set of all tariff arcs on a path $p \in P_k$	34
F_p	Set of all fixed cost arcs on a path $p \in P_k$	34
$\delta_a(d)$	Cost of routing a demand d on an arc $a \in F$	34
$\rho_a(d)$	Pricing strategy for a demand d on a tariff arc $a \in T$	34
$\pi_p(d)$	Revenue for a path p and a demand d	34
$\gamma_p(d)$	Fixed cost for a path p for a demand d	34
$\lambda_p(d)$	Length of a path p for a demand d ($\lambda_p(d) = \gamma_p(d) + \pi_p(d)$).	34
u_k	Shortest path from s_k to t_k with highest total fixed cost.	36
l_k	Shortest path from s_k to t_k with lowest total fixed cost.	36
$U_k(d)$	Upper bound on revenue for a client k with a demand d	36
u_{ij}	Shortest path from i to j in G with highest total fixed cost.	43
l_{ij}	Shortest path from i to j in G with lowest total fixed cost.	43

Acknowledgements

First of all, I would like to thank my promotor, Stan van Hoesel. I am grateful for his professional expertise, which he was always more than willing to share and of which I have benefited greatly. Besides our professional relationship, I have enjoyed getting to know him on a more personal level.

I am thankful for the financial support of France Télécom Research & Development, which allowed me to research a very interesting topic. Of all the people at France Télécom, I especially want to thank Mustapha Bouhtou for a fruitful working relationship that has its roots before this thesis. He was always available to discuss our research, be it by mail, telephone or in person. Thanks to all the other people at France Télécom Research & Development in Paris, who always welcomed me heartily when I came to visit them.

My colleagues here at Maastricht University have provided me with a very pleasant working environment, for which I would like to thank them all.

Special thanks to Alexander Grigoriev, Jean-Luc Lutton, Carlo Mannino, Gianpaolo Oriolo and Marc Uetz for their collaboration on parts of this thesis and to the readers who have helped eliminate mistakes in the draft version.

A thesis cannot be written without the possibility to enjoy extracurricular activities. My friends have provided these on a regular basis. I am most grateful for our late nights, even for those I hardly remember.

Special thanks to Mr. and Mrs. Kuiper, for their kindness and hospitality, both on land and offshore.

My family has been a source of great support, especially my parents. I thank them for their unconditional love and immense confidence in me.

My heart goes to Marguérita. I feel privileged to have her at my side and wish to thank her for her love and support by dedicating this thesis to her.

Anton F. van der Kraaij
Maastricht, September 2004

Chapter 1

Introduction

Until recently, due to the monopolistic nature of the telecommunication markets in the European Union and the United States of America, a client did not have a choice of operator for his national or international communications. For his national communications he was restricted to the operator of his own country or state. That same state monopoly transmitted his international communications to the operator in the destination state. The prices the different state monopolies charged for transmitting their clients' communications were not determined by free market principles.

Both in the United States and the European Union, several laws (see EU Directive [27] and US Act [62]) aimed for the deregularisation of national telecommunication markets. With the opening of the national markets for foreign operators and the advent of mobile telephony, the rules of the game changed drastically. The number of operators in each country has increased significantly. A client now has a choice of operators, both for his fixed and mobile telecommunications. Similarly, each operator faces competition for transmitting the communications of clients, both where national and international communications are concerned. In this changed market, the pricing strategy each operator has for transmitting his clients' communications, is an important aspect of his overall business strategy. It is in light of this view that this research is accomplished.

1.1 A Stackelberg Pricing Game

Consider a network defined by a set of nodes and a set of arcs connecting these nodes. Several clients are to be routed on the network. Whenever a client is routed on an arc, he faces the cost of using that particular arc. Each client wishes to route his demand from source to destination in the network, and wishes to do so at minimum cost. If the cost of routing on all arcs is given beforehand, the problem consists of solving for a shortest path in the network for each client.

We introduce an agent in the network, the leader, who can determine the cost of routing each client on a subset of all arcs in the network, i.e. the tariff arcs. Whenever a client takes a tariff arc, the associated cost is transferred as revenue to the leader.

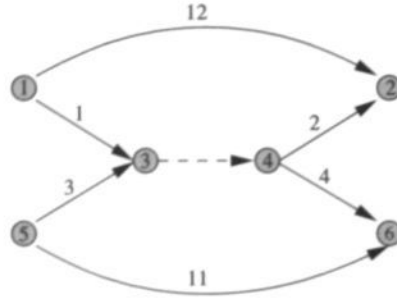


Figure 1.1: The NTP with two clients and a single tariff arc.

The leader wishes to maximize his revenue from clients taking tariff arcs and plays a sequential, two-stage game with the clients. In the first stage, he sets the tariffs on the tariff arcs, taking into account the reaction of all clients. Once the leader has set the tariffs, the cost of routing on all arcs in the network is known and the clients are faced in the second stage of the game with a shortest path problem in the network. The game played by the leader and the clients is the network tariffication problem, and is an instance of a Stackelberg game [59].

The Network Tarification Problem

The network tariffication problem (NTP) involves two noncooperative groups: tariff setting agents and tariff following clients. We focus on one agent: the leader. We are given a network, where the leader can set nonnegative tariffs on a subset of all arcs in the network: the tariff arcs. The other arcs in the network are the fixed cost arcs. The fixed cost arcs are owned by other agents in the network, whose tariffs are known a priori. The cost of routing a client's demand on a fixed cost arc is thus given. Each client wishes to route a certain amount of flow (a commodity) on a path connecting two vertices. We assume that all demands of all clients are known a priori. Given the tariffs as set by the leader, each client selfishly selects a path with minimum total cost to route his demand. Such a path can involve tariff and fixed cost arcs. The cost on the tariff arcs of that path is transferred as revenue to the leader. The objective of the leader is to maximize his individual total revenue.

Example 1.1. Consider the NTP for the network given in Figure 1.1. We are dealing with two clients with unit demand. The first client (client *A*) wishes to route his demand from node 1 to node 2. The second client (client *B*) wishes to route his demand from node 5 to node 6. The leader has to set the tariff for a single tariff arc, depicted by the dashed arc: The arc (3,4) with tariff v_1 per unit of demand. The other arcs in the network are the fixed cost arcs, whose cost per unit of demand is given.

Since both clients have unit demands, for each client the cost of routing on the tariff arc is v_1 . If we consider client *B*, we see that the leader can set a tariff of at most 4. For a tariff of $v_1 = 4$, client *B* takes the path (5,3,4,6) with cost $3 + 4 + 4 = 11$,

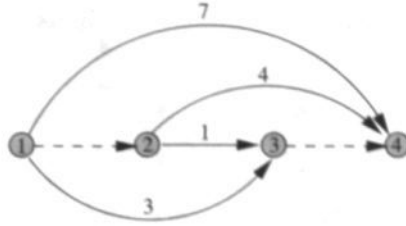


Figure 1.2: The NTP with a single client and two tariff arcs.

yielding a revenue of 4 to the leader. Moreover, for a tariff of $v_1 = 4$, client A chooses to take the path $(1, 3, 4, 2)$ with cost of $1 + 4 + 2 = 7$, yielding an additional revenue of 4 to the leader. Total revenue to the leader is thus 8. For a tariff lower than 4, clients A and B still take the tariff arc in the network, but their paths yield a total revenue to the leader that is smaller than 8. This is not in the best interest of the leader, who will thus never choose a tariff lower than 4. For a tariff higher than 4 on the tariff arc, client B will not take the path going through the tariff arc, but take a path that is shorter (the path $(5, 6)$), yielding no revenue to the leader. Client A will take the tariff arc, as long as the tariff is not higher than 9. The leader will however never set the tariff between 4 and 9, since this is not in his best interest: He can increase his revenue from client A by increasing the tariff until $v_1 = 9$. The optimal solution for this example is a tariff of $v_1 = 9$, yielding a revenue of 9 to the leader.

Example 1.1 shows the contradictory interests of the leader and the clients: The leader wishes to maximize his revenue, while the clients wish to take the path with minimum cost in the network. Once the tariffs are set by the leader, the clients can choose to take a path in the network with no tariff arcs. Note that whenever two paths exist in the optimal solution with the same cost for the client, ties are broken in favor of the leader, who can always decrease the cost of a path by $\varepsilon > 0$.

Example 1.2. Consider the NTP for the network given in Figure 1.2. We are dealing with a single client with a demand of 2 from source node 1 to target node 4. The leader on the network has to set the tariffs for the tariff arcs depicted by the dashed arcs, i.e. the arcs $(1, 2)$ with tariff v_1 per unit of demand and $(3, 4)$ with tariff v_2 per unit of demand. The other arcs are the fixed cost arcs, whose cost per unit of demand is given.

Since the client has a demand of 2, the cost of routing his demand on the arc $(1, 2)$ is given by $v_1 \cdot 2$. Likewise, the cost of routing his demand on the arc $(3, 4)$ is given by $v_2 \cdot 2$. The client has four different paths to choose from: the path $p_1 = (1, 2, 3, 4)$ taking two tariff arcs, the paths $p_2 = (1, 2, 4)$ and $p_3 = (1, 3, 4)$, taking only one of the two tariff arcs, and the path $p_4 = (1, 4)$ taking no tariff arcs. In the optimal solution, given the tariffs as set by the leader, the client takes the shortest path from 1 to 4 that yields the highest revenue to the leader.

Suppose that the path p_2 is the shortest path in the optimal solution. Setting $v_1 = 3$ and v_2 large enough (a tariff higher than 4 in this case) maximizes the revenue for the leader on the path p_2 . The tariff $v_2 > 4$ is necessary to ensure that the shortest

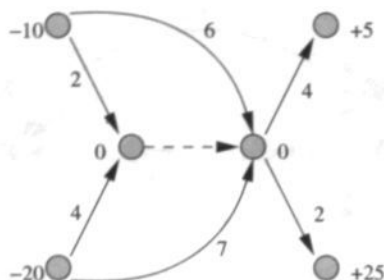


Figure 1.3: A simple freight tariff setting problem.

path for the client is indeed the path p_2 . For the solution $v_1 = 3$ and $v_2 > 4$, the cost of the path p_2 is $7 \cdot 2 = 14$ and the revenue to the leader is $3 \cdot 2 = 6$. Likewise, if we suppose that the path p_3 is the shortest path in the optimal solution, the tariffs on the arcs must be set to $v_1 > 3$ and $v_2 = 4$. The cost of the path is 14, and the revenue to the leader is equal to $4 \cdot 2 = 8$. The optimal solution however is for the client to take the path p_1 , given the tariffs $v_1 = 2$ and $v_2 = 3$. For this solution, the cost of the path p_1 is $6 \cdot 2 = 12$ and the revenue to the leader is $5 \cdot 2 = 10$.

Example 1.1 and Example 1.2 are both instances of the network tariffication problem. As we shall see, their complexity is however very different: An efficient algorithm to find the optimal solution of the network tariffication problem with a *single* tariff arc exists. It is highly unlikely¹ that an efficient algorithm exists to find the optimal solution of the network tariffication problem with *multiple* tariff arcs.

1.2 Applications

The network tariffication problem can be found in different industry sectors. Traditionally, most applications were in the transportation industry. Due to the deregulation discussed earlier, several applications are currently found in the telecommunications industry. We highlight some applications from these two industry sectors in this section.

1.2.1 Transportation Industry

Consider a freight transportation network. A shipper company is faced with the problem of transporting an amount of goods from origin nodes to destination nodes at minimum cost. The cost of transporting the goods between nodes is determined by the carriers. Each carrier is faced with the problem of determining these costs on the arcs of the network that are in his control. When the cost of routing is known for all carriers, the shipper company is faced with a standard transportation problem.

¹The NTP is NP-hard, even for a single client and multiple tariff arcs (see Roch, Savard, and Marcotte [53]).

Brotcorne, Labbé, Marcotte, and Savard [13] studied this problem and focused on one carrier, who has to set the prices for routing flow on the arcs in his control. The prices the other carriers in the network are asking for routing the flow on their arcs are given beforehand and are thus viewed as costs. The objective of the carrier is to maximize his revenue. See for example the network in Figure 1.3. For this network, the carrier has to determine the cost of routing on a single tariff arc (given by the dashed arc in the network). The shipper company is faced with two supply nodes and two demand nodes. The other nodes in the network are transshipment nodes. For the carrier, only two different tariffs are reasonable in this example. He either sets the tariff to 4, yielding a revenue of 40, or he sets the tariff to 3, yielding a revenue of 90. The optimal strategy for the carrier is thus to set the cost of routing on the tariff arc to 3.

Another application of the network tariffication problem is the highway toll optimization problem, as studied for example in Labbé, Marcotte, and Savard [41] and Brotcorne, Labbé, Marcotte, and Savard [14]. Consider a network of motorways linking several cities with each other and suppose that some of the motorways belong to a private company or local government (the leader). These motorways are subject to tolls. Several clients in the network wish to go from one city to another one at minimum cost. The problem of the leader is to maximize his revenue from the tolls in the network, taking into account the reaction of the clients to the tolls.

1.2.2 Telecommunications Industry

The optimization of the price for transmitting communication in a network can give the operator of a network a competitive advantage. Consider thus a telecommunications network where several operators are active, as given in Figure 1.4 for the international communications market. A node in this network represents access to the network of an operator. In the example of Figure 1.4, two operators are active in France, where each operator can provide access to all destinations in France. For this application of the network tariffication problem (the interconnections market), we focus on one operator, the leader. The interconnections between the network of the leader and the other operators are to be priced, in order to maximize the revenue of the leader. The clients wish to transmit their demands from source to destination in the network at minimum cost. In the optimal solution, the leader therefore routes the demand, for the clients served, on the cheapest path in the network. Although this route may go through the network of several operators, the client is only aware of the leader and the total cost of the path. Note that for the international interconnections market, an operator may be active in several countries.

The pricing of interconnections occurs not only at the international level, but whenever operators have to decide on the price of interconnecting their networks. It thus also occurs for example among the operators inside a country, who have to transmit (mobile) communications from one point in the country to another point.

The network tariffication problem is also encountered on the internet. The data is transmitted on the internet using packets, as determined by the Transmission Control Protocol (TCP) (see Stevens [60]). As such, data sent between two nodes in



Figure 1.4: Interconnections market.

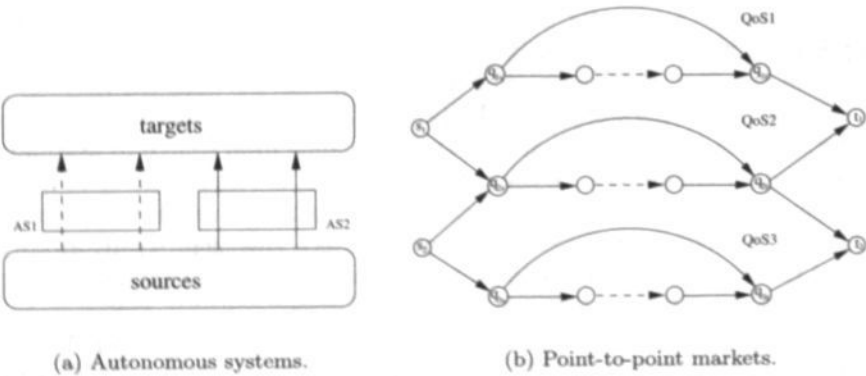


Figure 1.5: Applications of the network tariffication problem.

the network can take disjoint paths. However, as noted by Ben-Ameur, Michel, Liau, Geffard, and Gourdin [5], some operators (for example France Télécom), require the routing in the network to follow unique shortest paths, to better understand what happens in the network. Whenever an autonomous system (represented by some part of the network) has to transit data going through that network to other destinations, the data can enter the network at different points. Other operators have to pay a price for sending data through the autonomous system. The data flow can be modeled such that once it is routed through the network, it does not pass a second time. See for example Figure 1.5(a), where the clients have a choice of using two different autonomous networks with two entry points to route their traffic from source to destination.

Another application is in the field of point-to-point markets. Consider, for example, a telecommunications operator that is offering bandwidth capacity between two points A and B at a certain price. Other operators are also active in this market, whose prices for bandwidth capacity are known. The customer can select from different levels of Quality of Service (QoS) from each operator. The customer has an ordered preference for all QoS levels.

We can model this problem as an instance of the network tarification problem. Figure 1.5(b) shows a small example with two customers, represented by two commodities (s_1, t_1) and (s_2, t_2) . The operator has three QoS levels, represented by the subnetwork between the nodes q_{is} and q_{it} , where $i \in \{1, 2, 3\}$.

In this example, customer (s_1, t_1) is interested in two quality of service levels, namely QoS1 and QoS2, whereas customer (s_2, t_2) is interested in QoS2 and QoS3. The preference of each customer $k \in K$ with regard to each QoS level is given by the cost of the edge from the source s_k to the node q_{is} , $i \in \{1, 2, 3\}$. A smaller cost indicates a higher preference for the QoS level. The price the other operators are asking for the same QoS level is given by the cost on the arc (q_{is}, q_{it}) , $i \in \{1, 2, 3\}$. The revenue to the operator for each QoS level i , $i \in \{1, 2, 3\}$ is given by the tariff v_i on the tariff arcs (indicated by dashed arcs in the figure). The other fixed cost arcs represent the cost of opening the market to the operator. The objective of the operator is to maximize his revenue.

The networks in Figure 1.5(a) and Figure 1.5(b) have a special structure. This special case of the network tarification problem is discussed in Section 1.3.2.

1.3 Variations on the NTP considered

In this section, we introduce several variations on the network tarification problem studied in this thesis.

1.3.1 Pricing Strategy

At this point, we have only given examples of the network tarification problem where the pricing strategy used by the agents was linear in the demand d . Other pricing strategies may be available to the leader of the network. See Figure 1.6 for several

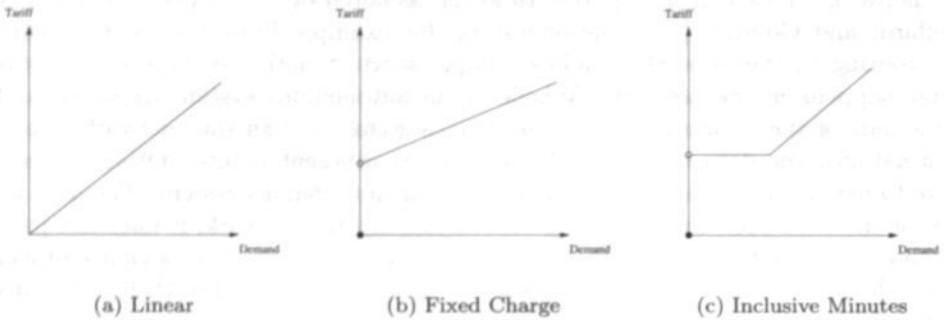


Figure 1.6: Several examples of pricing strategies.

pricing strategies studied in this thesis. Figure 1.6(a) shows the linear pricing strategy discussed until now.

Figure 1.6(b) shows a fixed charge pricing strategy. For this pricing strategy, the client pays a fixed price to be able to route his demand on an arc in the network. The client incurs furthermore a cost per unit of traffic. Such a pricing strategy is commonly used by telecommunication operators in the market for fixed telephony.

Another example of a pricing strategy is the *inclusive minutes pricing strategy*, illustrated in Figure 1.6(c). For this pricing strategy, the client pays a fixed price to route a predetermined amount of data on the network. If his data communications exceed this predetermined amount, he pays an additional price per unit of extra traffic. Telecommunications operators commonly use an inclusive minutes pricing strategy in the market for mobile telephony.

As we will see in Section 2.2, where we formally define all pricing strategies studied, other pricing strategies are also possible in the network. All pricing strategies studied are nonnegative, nondecreasing functions in the demand.

1.3.2 Network Structure

An important special case of the network tariffication problem is the problem where we assume that the path taken by any client utilizes at most one tariff arc. It is clear from the applications of autonomous systems and point-to-point markets, described in Section 1.2.2, that this network structure may indeed appear in practice. For these problems, we can thus restrict ourselves to special networks that resemble the situation of a town that is divided by a river. To cross the river, several possibilities exist, such as bridges, tunnels, or boats. Local government may decide to put tolls for each crossing on several of these transportation possibilities. This analogy is reflected in the name for this special case, the *river tariffication problem*.

This special case also occurs in the international interconnections markets. For some instances, the network structure may be such that we know a priori that the path of each client takes at most one tariff arc. Consider, for example, a particular instance of the international interconnections market, as given in Figure 1.4 on page

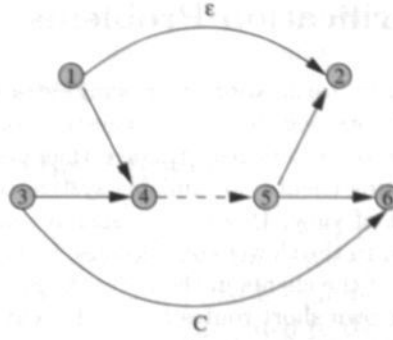


Figure 1.7: Worst case scenario for the ALL-SERVICE NTP.

22, where several operators offer connections to destinations inside the same country. If we focus on the market for one particular country, we know that all demands will be able to access the destinations inside that particular country through several non-sequential interconnections, for which the cost of routing must be determined. We can assume that it is not profitable for the client to enter the country more than once. The tariff arcs are thus in an $(s - t)$ -cut, for all demands from a node s to a node t .

1.3.3 All-service NTP

For the network tariffication problem, the clients are routed on the shortest path in the network. Thus, in the optimal solution, some clients may take a path consisting only of fixed cost arcs. These clients are thus served only by the competition. For the all-service network tariffication problem (ASNTP), the leader assumes that he knows beforehand all clients that must take one of the arcs in his control, i.e. all clients that must be serviced. An optimal solution to the all-service network tariffication problem consists of revenue maximizing tariffs on the tariff arcs such that, given these tariffs, each client is routed on the shortest path in the network, a shortest path that takes at least one of the tariff arcs of the leader.

Figure 1.7 shows an example where the gap between the optimal solution of the network tariffication problem and the all-service network tariffication problem may be arbitrarily large. For this network, the leader has a linear pricing strategy on one tariff arc (given by the dashed arc). Two clients wish to route their demand in the network. The first client has a unit demand from node 1 to node 2, with an upper bound on the cost of the path of $\varepsilon > 0$. The second client has a unit demand from node 3 to node 6, with an upper bound on the cost of the path of C , where C is arbitrarily large. The cost of routing on the other fixed cost arcs in the network is zero. The optimal solution for the network tariffication problem is to serve only the second client, yielding a revenue of C . For the all-service network tariffication problem, the optimal solution yields a revenue of 2ε .

1.4 Related Tarification Problems

In the transportation industry, tarification (or pricing) strategies can be used to distribute traffic over the network, e.g. to avoid congestion, or more generally in order to minimize some measure of system performance that does not coincide with the objectives of the participating clients. Minimizing system congestion is certainly desirable from a global point of view. However, in such a solution, a small fraction of the total traffic is sacrificed to the slower edge because it improves the congestion experienced by the majority of the clients in the network. Since very few drivers would be willing to sacrifice their own short routes for the benefit of others, this drawback has inspired Jahn, Möhring, and Schulz [35] to find traffic assignments that minimize system congestion subject to length constraints. In this context, the structural analysis of (Nash) equilibria versus system optima in networks with congestion has been heavily researched recently. The papers by Roughgarden [55], MacKie-Mason and Varian [47], and Cole, Dodis, and Roughgarden [20] are some examples.

Road pricing however also has a political aspect: how and when to introduce it. It may therefore be difficult to impose optimal or near-optimal tolls due to political reasons. This results in a congestion-pricing problem where it is not possible to set the tolls on all roads in the network. In that situation, the flow in the network is said to be (partially) unregulated. Liu and McDonald [45] and Roughgarden and Tardos [56] studied this particular case and analyzed the loss in network performance from this lack of regulation. As shown by Braess's paradox (see Braess [12]), careful consideration needs to be taken when planning (new) roads and toll points: In some cases removing roads in congested networks improves network performance. Braess's paradox suggests the following network design problem: given a network with latency functions on the edges and a traffic rate, which edges should be removed to obtain the best possible flow at Nash equilibrium? Since levying a sufficiently high tax on an edge in a network effectively removes it, taxation is at least as powerful as edge removal. Roughgarden [54] showed that designing networks such that congestion is minimized is NP-hard. This result is generalized in Cole, Dodis, and Roughgarden [19], where edges are not only considered for removal but also for optimal pricing. Verhoef [65] developed heuristic algorithms for the problem of determining the optimal toll points and corresponding toll levels.

The transportation industry is not the only sector that is interested in (congestion) pricing. Côté, Marcotte, and Savard [21] studied the tarification problem that is found in the field of yield management in the airline industry as one of the subproblems of the revenue optimization chain of an airline. Their problem fits in the framework where the leader has a small market share: The leader's capacity is limited, while the competition's capacity is assumed unlimited.

In the paper by Brotcorne, Labbé, Marcotte, and Savard [15], the network tarification problem is combined with the network design problem in telecommunications networks. They studied not only which tariffs to set on the tariff arcs, but also whether or not to open the tariff arcs for routing. For this problem, congestion does however not play a role, since the arcs in the network are uncappeditated. Bouhtou, Diallo, and Wynter [7] studied a pricing problem in telecommunication networks

with capacity constraints, where the prices on the tariff arcs are restricted to be Lagrangian multipliers for the capacity constraints. These prices will therefore be strictly positive only when capacity is reached.

1.5 Outline of the Thesis

In the next chapter, we formally introduce the network tarification problem and show that it is a special case of bilevel programming. We introduce a remodeling of the network, the shortest path graph model, that is equivalent to the original network.

The other chapters of the thesis are dedicated to the complexity, approximation and optimization of the network tarification problem. These chapters build on the notation and problem definitions described in the first part of Chapter 2.

In Chapter 3, we investigate the complexity of the problem studied. The network tarification problem and two special cases, the river tarification problem and the all-service network tarification problem, are shown to be \mathcal{NP} -hard. The remainder of the chapter identifies some polynomially solvable special cases of the network tarification problem. Chapter 4 is dedicated to approximation results for the network tarification problem. We propose approximation algorithms for the river tarification problem, and non-approximability results for the all-service network tarification problem. In Chapter 5 and Chapter 6 we describe several exact algorithms. In Chapter 5, we consider exact algorithms for a linear pricing strategy on the tariff arcs. In Chapter 6 these algorithms are generalized to allow for all pricing strategies studied. Moreover, the consequences of each pricing strategy on the revenue to the leader and for the clients is investigated.

Chapter 2

The Network Tarification Problem

This chapter formally introduces the network tarification problem studied. We model the network tarification problem as a special case of bilevel programming and define the different pricing strategies available. After defining our model and setting our notation, we briefly investigate the objective function of the leader. In Section 2.5, we propose a remodeling of the network, which is equivalent to the original network.

2.1 A Special Case of Bilevel Programming

Consider two players, A and B , who play a two-stage, sequential game. Player A chooses his strategy in the first stage of the game, taking into account the optimal strategy of player B . In the second stage, player B responds to player A 's strategy. Such a game is an example of a Stackelberg game (see Stackelberg [59], Tirole [61] and Varian [64]). The player with the first-move option is usually called the leader, whereas the second player is usually called the follower. Bilevel programming provides an adequate framework for such a sequential game.

The bilevel programming problem is a hierarchical optimization problem where the constraints are defined by a second (parametric) optimization problem (Marcotte and Savard [48], Vicente [67]). Denote the decision vector of the leader by x and the decision vector of the follower by y . A bilevel program is given by:

$$\begin{array}{ll} \min_{x,y} & \mathcal{F}(x,y) \\ \text{s.t.} & (x,y) \in X \\ & y \in \mathcal{S}(x). \end{array} \quad (2.1)$$

For the bilevel program in (2.1), $\mathcal{F}(x,y)$ represents the objective function of the leader, while $\mathcal{S}(x)$ represents the set of optimal solutions for the follower in terms of the leader's decision vector x . It is a mathematical program parameterized in the leader's vector x :

$$\begin{array}{ll} \arg \min_y & f(x, y) \\ \text{s.t.} & (x, y) \in Y. \end{array} \quad (2.2)$$

The optimization problem of the leader in (2.1) is referred to as the upper level or leader's problem, while (2.2) corresponds to the lower level or follower's problem. In the literature, several classes of bilevel programs are defined. When both objective functions are linear, the problem is identified as a linear-linear bilevel program. Similarly, a bilinear-linear bilevel program is a bilevel program where the leader's objective function is bilinear, and the follower's objective function is linear. Analogously, we can define linear-quadratic bilevel programs, and the quadratic bilevel program where both objective functions are quadratic. The difference between bilinear and quadratic functions is that bilinear functions are linear with respect to each of two variables.

Combining the upper level and lower level in one formulation yields the following bilevel program:

$$\begin{array}{ll} \min_{x,y} & \mathcal{F}(x, y) \\ \text{s.t.} & (x, y) \in X \\ & y \in \arg \min_{y'} f(x, y') \\ & \text{s.t.} \quad (x, y') \in Y. \end{array} \quad (2.3)$$

Since the original formulation of a bilevel program by Bracken and McGill [11], several authors studied bilevel programming intensively. The general linear-linear bilevel program has been shown to be \mathcal{NP} -hard by Jeroslow [36] and strongly \mathcal{NP} -hard by Hansen, Jaumard, and Savard [33]. Even checking for local optimality in a linear-linear bilevel problem is an \mathcal{NP} -hard problem, as shown by Vicente, Savard, and Júdice [66]. Several algorithms using different methods have been proposed to solve bilevel programming. Extreme point algorithms for linear bilevel programming problems were proposed by Candler and Townsley [17], Bialas and Karwan [6]. These algorithms use the property proved by Bard [3], Bialas and Karwan [6] that at least one global optimal solution is attained at an extreme point of the set of relaxed feasible solutions $\Omega = \{(x, y) : (x, y) \in X, (x, y) \in Y\}$, when this set is bounded. For the linear-linear bilevel problem, branch and bound algorithms have been proposed by Bard and Falk [4] and Fortuny-Amat and McCarl [28]. Classical examples of descent methods are the algorithms proposed by Savard and Gauvin [57] and Kolstad and Lasdon [40]. The books by Luo, Pang, and Ralph [46] and Shimizu, Ishizuka, and Bard [58] are devoted to the theory and applications of bilevel programming. For additional references on bilevel programming, we refer the reader to Dempe [24], Vicente and Calamai [68] who have compiled annotated bibliographies on this subject containing more than one hundred references.

Bilevel Programs and the NTP

As shown by Labbé, Marcotte, and Savard [42], Marcotte and Savard [49], bilevel programs provide a very natural framework for problems that arise whenever tariffs,

tolls or taxes are imposed on a set of commodities or activities. Let the vector x denote the level of taxed activities and the vector y the level of untaxed activities. The tax vector associated with the activity vector x is denoted by τ . The leader seeks to maximize the revenues raised from taxation, knowing that the follower will react to his tax levels by trying to minimize his costs. This yields the following bilevel program:

$$\begin{aligned} \max_{\tau} \quad & \mathcal{F}(\tau, x, y) \\ \text{s.t.} \quad & (x, y) \in \arg \min_{x', y'} f(\tau, x', y') \\ & \text{s.t.} \quad (x', y') \in Y, \end{aligned} \quad (2.4)$$

where $\mathcal{F}(\tau, x, y)$ is the objective function of the leader and $f(\tau, x, y)$ the objective function of the follower. A bilevel program is generally written in vertical form. For the bilevel program in (2.4) this yields:

$$\begin{aligned} \max_{\tau} \quad & \mathcal{F}(\tau, x, y) \\ \min_{x, y} \quad & f(\tau, x, y) \\ \text{s.t.} \quad & (x, y) \in Y. \end{aligned} \quad (2.5)$$

Consider the network tarification problem with a linear pricing strategy on the tariff arcs and a single client. For the bilevel program in (2.5), let x denote the flow on the tariff arcs and y the flow on the fixed cost arcs. The cost of routing on the fixed cost arcs is denoted by the cost vector c . We can then write (2.5) as follows:

$$\begin{aligned} \max_{\tau} \quad & \tau^{\top} x \\ \min_{x, y} \quad & \tau^{\top} x + c^{\top} y \\ \text{s.t.} \quad & A_1 x + A_2 y = b. \\ & x, y \geq 0 \end{aligned} \quad (2.6)$$

For the bilevel program in (2.6), A_1 , respectively A_2 , is the node-arc incidence matrix for the tariff, respectively fixed cost, arcs. The vector b is the supply/demand vector for the client in the network. For the formulation in (2.6), we have written the network tarification problem as a special case of bilevel programming. Both objective functions are bilinear in this formulation. It is possible to write the bilevel program (2.6) as a single-level program. Assuming the polyhedron $\{(x, y) : A_1 x + A_2 y = b, x, y \geq 0\}$ is bounded, and the polyhedron $\{(x, y) : A_2 y = b, y \geq 0\}$ is nonempty, we can replace the lower level with its optimality conditions (primal and dual feasibility and primal and dual objective equality, see Chvátal [18]). This yields the following (nonlinear) single-level formulation, where λ represents the variables of the dual problem:

$$\begin{aligned}
& \max_{\tau, \lambda, x, y} \quad \tau^\top x \\
& \text{s.t.} \quad A_1 x + A_2 y = b. \\
& \quad A_1^\top \lambda \leq \tau \\
& \quad A_2^\top \lambda \leq c \\
& \quad \tau^\top x + c^\top y = b^\top \lambda \\
& \quad x, y \geq 0
\end{aligned} \tag{2.7}$$

2.2 Pricing Strategies

In this section, we formally define the different pricing strategies available to the agents of the network that we will study in this thesis, as described in Section 1.3.1. Denote the set of arcs in the network by A . The set A is partitioned into the set of tariff arcs T and the set of fixed cost arcs F . For each arc $a \in A$, the cost of routing a demand d is determined by its pricing strategy, denoted by $\rho_a(d)$. All pricing strategies $\rho_a(d)$ studied are piecewise linear (not necessarily continuous) functions. All pricing strategies are nonnegative, nondecreasing functions in d . The pricing strategies that we will define besides a linear pricing strategy are commonly used in telecommunication networks. All pricing strategies studied limit the number of linear pieces in the function $\rho_a(d)$ to at most 3.

Linear Pricing Strategy

For a linear pricing strategy, the cost of routing a client with demand d on the arc $a \in A$ is determined by a variable unit cost v_a only. The pricing strategy $\rho_a(d)$ for an arc $a \in A$ is defined as follows:

$$\rho_a(d) = v_a d. \tag{2.8}$$

Fixed Charge Pricing Strategy

In case of a fixed charge pricing strategy, clients pay a fixed price to be able to route their demands on the network. Furthermore, for each unit of traffic, the client incurs a nonnegative cost per unit of traffic. Hence, for a fixed charge pricing strategy, the pricing strategy on each arc $a \in A$ is given by:

$$\rho_a(d) = \begin{cases} 0 & \text{if } d = 0 \\ f_a + v_a d & \text{if } d > 0. \end{cases} \tag{2.9}$$

Inclusive Minutes Pricing Strategy

A small adaptation of the fixed charge pricing strategy is the inclusive minutes pricing strategy, where the client pays a fixed price for a certain predetermined amount of traffic. This amount of traffic is assumed to be known a priori and (possibly) different for each tariff arc. We denote it by d_a^1 for each arc $a \in A$. If the client wishes to

route more than d_a^1 on the arc $a \in A$, he pays an additional cost per unit of extra traffic:

$$\rho_a(d) = \begin{cases} 0 & \text{if } d = 0 \\ f_a^1 & \text{if } 0 < d \leq d_a^1, \\ f_a^2 + v_a^2 d & \text{if } d > d_a^1. \end{cases} \quad (2.10)$$

Piecewise Linear Pricing Strategy

In case of a piecewise linear pricing strategy, the client pays a fixed price and a variable cost per unit of traffic, for each block of traffic. With this pricing strategy, the function $\rho_a(d)$ for a client with demand d on the tariff arc $a \in A$ is:

$$\rho_a(d) = \begin{cases} 0 & \text{if } d = 0 \\ f_a^1 + v_a^1 d & \text{if } 0 < d \leq d_a^1, \\ f_a^2 + v_a^2 d & \text{if } d_a^1 < d \leq d_a^2, \\ f_a^3 + v_a^3 d & \text{if } d > d_a^2. \end{cases} \quad (2.11)$$

We assume the values of d_a^1 and d_a^2 are known a priori for each arc $a \in A$.

Staircase Pricing Strategy

With a staircase pricing strategy, the client pays a fixed price for each predetermined block of traffic in the network. For this pricing strategy, the function $\rho_a(d)$ for the arc $a \in A$ is given by:

$$\rho_a(d) = \begin{cases} 0 & \text{if } d = 0 \\ f_a^1 & \text{if } 0 < d \leq d_a^1, \\ f_a^2 & \text{if } d_a^1 < d \leq d_a^2, \\ f_a^3 & \text{if } d > d_a^2. \end{cases} \quad (2.12)$$

As is the case with the inclusive minutes pricing strategy, we assume the values of d_a^1 and d_a^2 are known a priori for each arc $a \in A$.

Parametric Pricing Strategy

When dealing with a parametric pricing strategy, there is a single parameterized tariff, say $v \geq 0$, for all arcs. The pricing strategy on each arc $a \in A$ is defined as:

$$\rho_a(d) = (\alpha_a + \beta_a v)d. \quad (2.13)$$

A parametric pricing strategy has furthermore an interesting special case, where $\alpha_a = 0$ and $\beta_a = 1$ for all arcs $a \in T$. In that case, the tariffs are restricted to be equal on all tariff arcs. We refer to this type of pricing strategy as the uniform pricing strategy.

Pricing Strategy on the Fixed Cost Arcs

As stated previously, the cost of routing a demand d on a fixed cost arc $a \in F$ is known a priori. This cost is determined by the pricing strategy on the fixed cost arc, as set by the other agents in the network. As such, it can be any of the previously introduced pricing strategies. To emphasize that this pricing strategy is known a priori, we will denote the cost of routing a demand d on a fixed cost arc $a \in F$ by $\delta_a(d)$. This implies that $\delta_a(d)$ is a nonnegative, nondecreasing function in d for all arcs $a \in F$.

2.3 Model

The underlying structure of the network tarification problem (NTP) is a directed graph $G = (N, A)$, where A is partitioned into a set of tariff arcs T and a set of fixed cost arcs F . The clients form a set of commodities K , where each client $k \in K$ has a demand d_k from node s_k to node t_k . The tariff arcs belong to the leader and incur a toll for routing a client's demand. The fixed cost arcs are owned by other agents in the network, whose pricing strategy is given beforehand. We denote the cost of routing a demand d on an arc $a \in F$ by $\delta_a(d) \geq 0$. The pricing strategy on a tariff arc $a \in T$ is referred to by the nonnegative function $\rho_a(d)$, where d is the demand to be routed on the tariff arc. We define for a commodity $k \in K$ the set of all possible paths from source to target by P_k . Denote by T_p , respectively F_p , the set of tariff arcs, respectively the set of fixed cost arcs, on a path $p \in P_k$. The revenue associated with a path p for a demand d is defined by

$$\pi_p(d) := \sum_{a \in T_p} \rho_a(d), \quad (2.14)$$

whereas the fixed cost associated with the path p is given by:

$$\gamma_p(d) := \sum_{a \in F_p} \delta_a(d). \quad (2.15)$$

The parameters of the pricing strategies (tariffs) on the arcs of T are determined such that the total revenue of the leader is maximized. The clients on the network route their demands from source to destination according to the shortest path with respect to total cost, where the total cost of a path is defined as the sum of all tariffs and fixed costs on the arcs of the path: Define by $\lambda_p(d) = \gamma_p(d) + \pi_p(d)$, the length of a path p . Note that whenever the demand is fixed, $\lambda_p(d)$ is a piecewise linear, non-decreasing function in the tariffs. To ensure that the leader's revenue is bounded from above, we assume the following.

Assumption 2.1. *For each client, there always exists a path in the network from source to destination that uses none of the tariff arcs.*

We also suppose that whenever the client has a choice among multiple shortest paths with the same total cost but with different revenues for the leader, the client takes the shortest path that is most profitable to the leader.

Assumption 2.2. *For fixed tariffs, a client always takes the shortest path in the network that is most favorable to the leader.*

We can justify this assumption, by noting that the leader can always decrease the cost of the shortest path by $\varepsilon > 0$. In Section 2.2 we introduced all pricing strategies studied in this thesis that the leader can apply on the tariff arcs. These are pricing strategies that are commonly used among telecommunications operators and that have the following property:

Assumption 2.3. *The pricing strategy on a tariff arc is given by a nonnegative, nondecreasing function in the demand.*

The pricing strategies are non-discriminatory (i.e., all clients face the same pricing strategy for the same arc). We will furthermore consider networks in telecommunications, for which the amount of capacity available on the arcs in the network is such that all demand can be satisfied. We may therefore consider our networks to be uncapacitated:

Assumption 2.4. *The capacity on all arcs in the network is unlimited.*

Note that Assumption 2.4 does not imply that the network tariffication problem is separable for all clients. The clients are linked through the tariffs.

The network tariffication problem can be defined by the following bilevel program, which holds for all pricing strategies studied:

$$\begin{aligned} \max \quad & \sum_{k \in K} \pi_{p_k^*}(d_k) \\ \text{s.t.} \quad & p_k^* \in \arg \min_{p \in P_k} \lambda_p(d_k) \quad \forall k \in K \end{aligned} \quad (2.16)$$

For the bilevel program in (2.16), at the upper level the revenue of the leader is maximized, while at the lower level the clients take the shortest path, given the tariffs determined at the upper level. Since the demands are known a priori, objective functions of both the leader and the clients in (2.16) are linear in the tariffs and hence we are dealing with a so-called linear-linear bilevel program. Furthermore, since for each client the number of paths in the network may be exponential, the bilevel formulation in (2.16) may have an exponential number of constraints. Labbé et al. [41] defined a bilevel programming formulation using arc-variables, where the number of constraints is not exponential (see also Chapter 5). Their formulation is a generalization of the bilevel program given in (2.6), allowing for multiple clients. Both objective functions (lower and upper level) are bilinear in the formulation.

2.4 Investigating the Leader's Revenue

Labbé et al. [41] have shown that it is possible to obtain an upper bound on the revenue to the leader. For a client $k \in K$ with demand d_k from source node s_k to

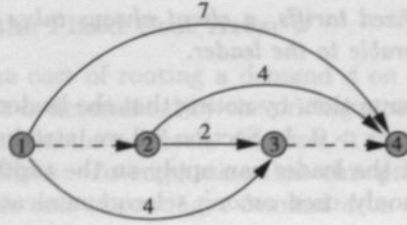


Figure 2.1: Upper bound on revenue is not reached.

target node t_k , denote the shortest path from s_k to t_k using only fixed cost arcs by the path $u_k \in P_k$. Denote the shortest path (possibly using tariff arcs) when all tariffs are set to zero by the path $l_k \in P_k$. Consequently, for a client $k \in K$, $\gamma_{u_k}(d_k)$ is an upper bound, and $\gamma_{l_k}(d_k)$ is a lower bound on the cost of a path from s_k to t_k .

Proposition 2.1 (Labbé et al. [41]). *Given is $u_k \in P_k$ and $l_k \in P_k$ for a client $k \in K$. An upper bound on the revenue for the leader from each client $k \in K$ is $\mathcal{U}_k(d_k) = \gamma_{u_k}(d_k) - \gamma_{l_k}(d_k)$.*

Proof. The proof follows directly from the definitions of u_k and l_k . See also Labbé et al. [41]. \square

Labbé et al. [41] showed that the upper bound $\mathcal{U}_k(d_k)$ of Proposition 2.1 is not necessarily reached in the optimal solution. To see this, consider the network tarification problem in the following example.

Example 2.1. We are dealing with a single client with a demand of 2 from source node 1 to target node 4 for the network given in Figure 2.1. The leader on the network has to set the tariffs for the tariff arcs depicted by the dashed arcs, i.e. the arcs (1, 2) with tariff v_1 per unit of demand and (3, 4) with tariff v_2 per unit of demand.

In Example 2.1, the leader uses a linear pricing strategy on the tariff arcs. An upper bound on the cost of the client's path from node 1 to node 4 is given by the cost of the shortest path from node 1 to node 4 using only fixed cost arcs, i.e. the path $u_1 = (1, 4)$ with total fixed cost $7 \cdot 2$. A lower bound on the cost of the client's path is given by the cost of the shortest path (possibly using tariff arcs) when all tariffs are set to zero, i.e. the path $l_1 = (1, 2, 3, 4)$, using both tariff arcs, with a total fixed cost of $2 \cdot 2$. Hence, an upper bound on the revenue for the leader is given by $14 - 4 = 10$.

The client has four different paths to choose from: the paths $p_1 = (1, 2, 4)$ and $p_2 = (1, 3, 4)$, taking only one of the two tariff arcs, and the paths $u_1 = (1, 4)$ and $l_1 = (1, 2, 3, 4)$. Suppose the client takes the path p_1 in the optimal solution. It is easy to see that setting $v_1 = 3$ and v_2 large enough (a tariff higher than 3 in this case) maximizes the revenue for the operator on the path p_1 , while ensuring that the shortest path for the client is indeed p_1 . Note that actually, for the solution $v_1 = 3, v_2 > 3$, the path p_1 and u_1 have the same cost for the client. In that situation,

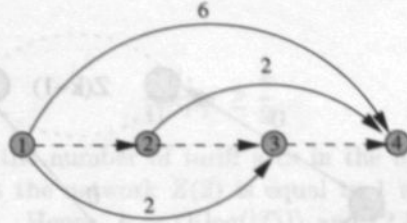


Figure 2.2: Optimal tariffs may be negative.

we apply Assumption 2.2 and break the tie in favor of the leader. For tariffs $v_1 = 3$ and $v_2 > 3$, the revenue for the operator is $3 \cdot 2$ and the upper bound on the cost of the shortest path from 1 to 4 is reached. For the path p_2 , the same reasoning shows that the maximal revenue for the operator on the path is also $3 \cdot 2$. It is easy to verify that the optimal pricing strategy is to set the tariffs to $v_1 = v_2 = 2$, inducing the client to take the path l_1 yielding a revenue of $4 \cdot 2$ to the leader. Both the upper bound on the revenue for the leader (given by $7 \cdot 2 - 2 \cdot 2 = 10$) as the upper bound on the cost for the client (which is equal to $7 \cdot 2 = 14$) are not achieved in the optimal solution.

The pricing strategies studied in this thesis are all nonnegative, nondecreasing functions in the demand d (see Section 2.2 and Assumption 2.3). The following example by Labbé et al. [41] shows that it may be profitable for the leader to allow for negative tariffs. In the transportation industry, these negative tariffs can be viewed as subsidies on roads in the network.

Example 2.2. See the network described by Labbé et al. [41], given in Figure 2.2. We are dealing with a single client with unit demand from source node 1 to target node 4. The leader on the network has to set the tariffs for the tariff arcs depicted by the dashed arcs, i.e. the arcs (1,2), (2,3) and (3,4), with tariffs v_1 , v_2 , and v_3 , per unit of demand.

For tariffs $v_1 = v_3 = 4$, $v_2 = -2$, the revenue for the leader is 6, whereas nonnegative tariffs will allow for a revenue of at most 4.

Recently, Roch et al. [53] have given us more insight into the quality of the upper bound on the revenue defined in Proposition 2.1. Their example shows that the upper bound on the revenue can be a factor of $\mathcal{O}(\log(|T|))$ far from the optimum. A simplified version of their example follows.

We define the single-commodity networks $Z(k)$, $k \in \{2, \dots\}$ recursively. The network $Z(2)$ is given by two nodes, one tariff arc and one fixed cost arc, as depicted in Figure 2.3(a). In Figure 2.3(a), the tariff arc is given by the dashed arc, while the cost of the fixed cost arc is 2. A single client is active in the network, who wishes to route his demand from source s to target t . We construct network $Z(k)$, consisting of two copies of $Z(k-1)$ and two extra nodes, corresponding to the source s and target t for the only client in the network. This client has unit demand. The nodes s and t and the two subnetworks $Z(k-1)$ are linked by fixed cost arcs as depicted in Figure 2.3(b).

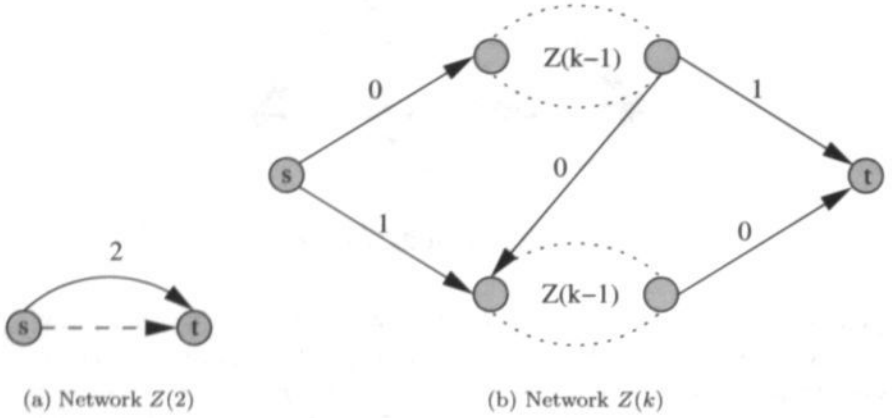


Figure 2.3: Upper bound on revenue $\mathcal{O}(\log(|T|))$ far from the optimum.

For a network $Z(k)$, a lower bound on the cost of a path from s to t is given by the shortest path (possibly taking tariff arcs) when the tariffs are set to zero. It is clear that the cost of such a path in $Z(2)$ is zero. Suppose such a zero cost path exists for $Z(k-1)$. In $Z(k)$ the client can either take a path going through the two subnetworks $Z(k-1)$ or the client can take one of the two paths going through only one subnetwork $Z(k-1)$. The shortest path (possibly taking tariff arcs) is the path going through the two subnetworks $Z(k-1)$ with zero total fixed cost.

The upper bound on the cost of a path from s to t in the network $Z(k)$ is given by the shortest path using only fixed cost arcs with total fixed cost k : For the network $Z(2)$ this is clear. Suppose the shortest path using only fixed cost arcs in the network $Z(k-1)$ has length $k-1$. In $Z(k)$ the client either takes a path going through the two subnetworks $Z(k-1)$ or the client takes one of the two paths going through only one subnetwork $Z(k-1)$. Since the cost of a shortest path using only fixed cost arcs through the two subnetworks $Z(k-1)$ has length $2(k-1)$, the shortest path in the network $Z(k)$ goes through only one of the subnetworks $Z(k-1)$ and has length k .

The value of the optimal revenue to the leader for the network $Z(k)$ is bounded by 2: For the network $Z(2)$ this is clear. Suppose the optimal revenue for the network $Z(k-1)$ is bounded by 2. In the network $Z(k)$ the client has three paths to choose from. If the client takes one of the two paths through only one of the subnetworks $Z(k-1)$, the revenue to the leader is at most 2 from one of the subnetworks $Z(k-1)$ on the path. If the client takes the path through both subnetworks $Z(k-1)$, the two fixed cost arcs with unit cost bound the revenue from each subnetwork $Z(k-1)$ on the path to at most 1. Hence, the revenue on the path through both subnetworks $Z(k-1)$ is at most 2.

Summarizing, for a network $Z(k)$ the upper bound as defined in Proposition 2.1 is equal to k while the optimal revenue from the network $Z(k)$ is bounded by 2. Denote the upper bound on the revenue for the single client in the network by \mathcal{U}_1 ,

and consider the ratio:

$$\frac{u_1}{\Pi^{OPT}} \geq \frac{k}{2}. \quad (2.17)$$

For the network $Z(k)$, the number of tariff arcs in the network is 2^{k-2} , since the number of tariff arcs in the network $Z(2)$ is equal to 1 and doubles for each new network that is created. Hence, $k = \mathcal{O}(\log(|T|))$ and (2.17) shows that the upper bound on the revenue can be a factor of $\mathcal{O}(\log(|T|))$ far from the optimum.

In order to gain some insight in the objective function of the leader, consider the network of Example 2.3.

Example 2.3. Consider the network in Figure 2.4. The leader is using a linear pricing strategy on the only tariff arc in the network, given by a dashed arc. The cost of routing on the fixed cost arcs is given for the nonzero fixed cost arcs. In the network 4 clients are active, each with unit demand. These are given by the commodities $(1, 2), (3, 4), (7, 8), (9, 10)$.

For the network of Example 2.3, the tariff on the tariff arc is given by v_1 , since we are dealing with a linear pricing strategy. For the leader, the revenue at a tariff of v_1 is given by:

$$\sum_{k \in K: u_k(d_k) \geq v_1} d_k v_1 \quad (2.18)$$

For this example, the objective function of the leader is plotted in Figure 2.5, showing that the objective function is discontinuous, nonconvex and piecewise linear. As noted by Labbé et al. [41], the objective function is upper semi-continuous, implying that there exists at least one optimal solution to the network tariffication problem.

2.5 The Shortest Path Graph Model

The networks considered for the network tariffication problem are quite often large networks. We propose a remodeling of the network using shortest paths, that is equivalent to the original network in the sense that the optimal solution in the original network has the same value as the optimal solution in the remodeled network. Generally, the new network is more compact.

2.5.1 Remodeling the Network

If for given tariffs a client will select the shortest path, say p , between the two end nodes of his commodity, then clearly, the subpaths of p are also shortest paths. Consider two tariff arcs $a_1 = (i_1, j_1)$ and $a_2 = (i_2, j_2)$ that appear consecutively on p . Then the subpath between j_1 and i_2 is a shortest path that contains only fixed cost arcs. Thus any path p with tariff arcs $a_i, i \in \{1, \dots, |T_p|\}$ taken by a client can be described as the path:

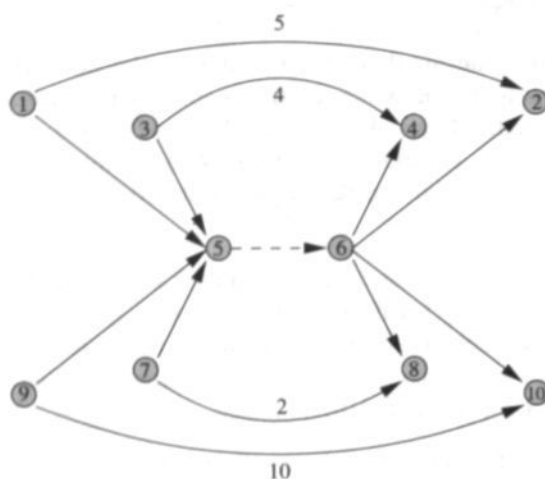


Figure 2.4: Network for Example 2.3.

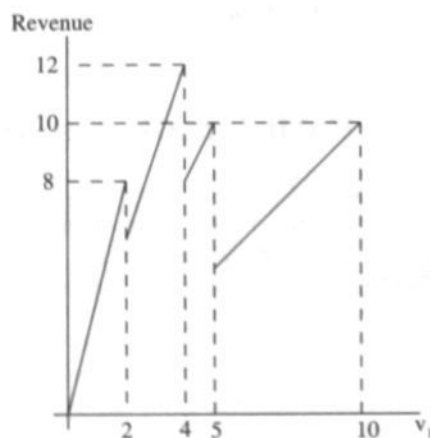
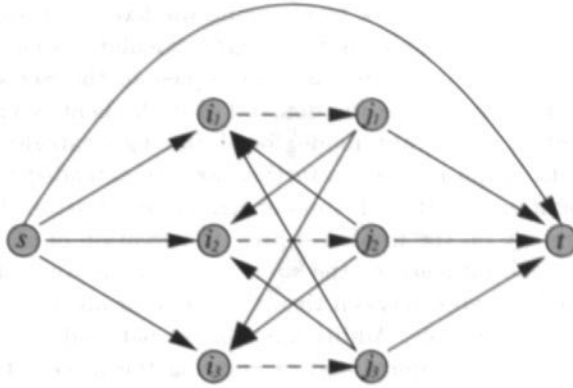


Figure 2.5: Shape of the leader's objective function.

Figure 2.6: Shortest Path Graph Model for $|T| = 3$.

$$p = \{sp_1, a_1, sp_2, a_2, \dots, sp_{|T_p|}, a_{|T_p|}, sp_{|T_p|+1}\}, \quad (2.19)$$

where sp_i , $i \in \{1, \dots, |T_p| + 1\}$ is a shortest subpath using only fixed cost arcs to or from a tariff arc on the path. Since such paths can be computed using the original data for all pricing strategies described in Section 2.2, we can construct a new graph model, in which this is actually done.

We define this graph model for a single client and a linear pricing strategy first. Subsequently, we argue that it holds for all pricing strategies studied and show how to extend it to multiple clients.

Consider the original graph $G = (N, A)$ with the tariff arcs in $T \subseteq A$. For a client with a demand d from s to t , we define the graph $G^* = (N^*, A^*)$ and the tariff arcs $T^* \subseteq A^*$. In this graph, the tariff arcs are copied from G as a set of separate arcs having no nodes in common. So, arcs with a common vertex are separated. Next, we construct the following fixed cost arcs. For two tariff arcs $a_1 = (i_1, j_1)$ and $a_2 = (i_2, j_2)$ we connect j_1 with i_2 , if there is a path in G that uses fixed cost arcs only. Similarly, we connect j_2 with i_1 . From the source s we construct arcs to each tail node of the tariff arcs, and from each head node of a tariff arc we construct an arc to the destination t , again only if paths exist using only fixed cost arcs in G . Finally, we construct a fixed cost arc in G^* from s to t . Any fixed cost arc in A^* has a cost equal to the length of the shortest path between its end vertices in G , using only fixed cost arcs in G . The new network is called the shortest path graph model (SPGM). (SPGM)

Example 2.4. Figure 2.6 shows the shortest path graph model of any network containing three tariff arcs for a commodity from s to t . The tariff arcs are given by the (dashed) arcs (i_1, j_1) , (i_2, j_2) and (i_3, j_3) . All other arcs are representations of the shortest path using only fixed cost arcs between each node. The cost of the arc is the cost of the corresponding shortest path in the original network between the two nodes. If no path exists between two nodes in the original network, the corresponding arc in the shortest path graph model is not present or has infinite cost.

The SPGM remodels the network by changing the fixed cost arcs of the network. It thus holds for all models where shortest paths calculations can be done a priori on the fixed cost arcs: the tariff arcs are just copied to the SPGM, including their pricing strategy. Since we assume the demand of each client is known beforehand, we can use the shortest path graph model for all pricing strategies.

The shortest path graph model can furthermore be extended to multiple clients, as shown next. Bouhtou, van Hoesel, van der Kraaij, and Lutton [10] noted that, for a linear pricing strategy on the fixed cost arcs, the creation of all the shortest path graph models has a special feature: the *inner graph* (consisting of the end vertices of the tariff arcs, and the arcs between them) is equal for all commodities and hence needs to be calculated only once. Additional shortest path calculations are necessary only for the arcs leaving the source and/or entering the target of each commodity. Thus, for a linear pricing strategy, one possibility to extend the shortest path graph model is to add arcs from all source nodes to all tariff arcs and from all tariff arcs to all target nodes. The other possibility is to extend it by creating a shortest path graph model for each client. This is even necessary for all pricing strategies described in Section 2.2, except the linear pricing strategy, since the path between two nodes of the inner graph is not necessarily the same for each commodity in that case. In Section 2.5.2, the size of the shortest path graph model will be reduced by deleting (redundant) arcs from the inner graph for each client. We therefore choose to extend the shortest path graph model to multiple clients by creating a shortest path graph model for each client, even in case of a linear pricing strategy.

Proposition 2.2. *The network of the shortest path graph model is equivalent to the original network.*

Proof. The shortest path graph model includes all shortest paths in the network, that are of the form (2.19), for all subsets $T_p \subseteq T$. Furthermore, the network tarification problem applied to each network has an optimal solution of the same value: if a path exists in the original network, then there exists a path in the shortest path graph model which is at least as good. Alternatively, if a path exists in the shortest path graph model, then a path with the same cost exists in the original network. \square

One of the main advantages of the shortest path graph model, is that several optimization methods developed for the original network can be applied to the SPGM. Often, these methods will perform better on the SPGM since it is a more compact representation of the original network, as we will illustrate in Chapter 5 for a linear pricing strategy and in Chapter 6 for the other pricing strategies studied. Furthermore, since the SPGM provides intuitive insight into the problem, its properties are useful for the proof of several theoretical results.

2.5.2 Reduction Methods

We can decrease the size of the shortest path graph model considerably by using reduction methods. The aim of these methods is to decrease the size of the SPGM even further and to limit the amount of potentially shortest paths for a commodity to an acceptable number. First, we describe some techniques with which we can remove

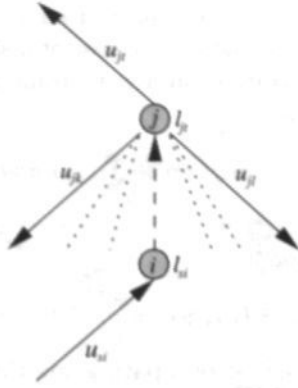


Figure 2.7: Detailed view of tariff arc.

arcs from the network. Second, we describe a dominance relationship for paths that reduces the set of potentially optimal paths. All reduction methods proposed are valid for all pricing strategies studied.

Arc Reductions

Denote by u_{ij} , the cost of the shortest path using only fixed cost arcs from node i to node j in G , i.e., u_{ij} is the length of the arc (i, j) in G^* . Let l_{ij} denote the cost of the shortest path from i to j in G , possibly using tariff arcs, when the tariffs are set to zero. We restrict ourselves to a single commodity, where node s represents the source node and node t the destination node. In Figure 2.7 we depict the values defined here: the u_{ij} are arc values, and the l_{ij} are node values.

Since for all $a \in T$ and all pricing strategies described in Section 2.2 it holds that $\rho_a(d) \geq 0$, we can say that l_{ij} is a lower bound for the cost of a path from i to j , and u_{ij} is an upper bound for the cost of a path from i to j taken by the client.

Proposition 2.3. *If $l_{jt} = u_{jt}$, then any optimal path from s to t using node j can use arc (j, t) : all other arcs leaving j can be removed.*

Proof. A lower bound on the cost of a path, possibly using tariff arcs, from j to t is l_{jt} . An upper bound is given by u_{jt} . If $l_{jt} = u_{jt}$, there is no room for taxation on any path from j to t . Thus, (j, t) is an optimal choice. \square

Proposition 2.4. *If $l_{si} = u_{si}$, then any optimal path from s to t using node i can use arc (s, i) : all other arcs entering i can be removed.*

Proof. Analogous to proposition 2.3. \square

Proposition 2.5. *Consider two tariff arcs: (i_1, j_1) and (i_2, j_2) . If $u_{j_1t} \leq u_{j_1i_2} + l_{j_2t}$, we can delete arc (j_1, i_2) .*

Proof. The lower bound on the cost of a path from j_1 to t taking the arc (j_1, i_2) is equal to $u_{j_1 i_2} + l_{j_2 t}$. An upper bound on the cost of a shortest path is given by $u_{j_1 t}$. Hence, there is no room for taxation on a path from j_1 to t using the arc (j_1, i_2) . Thus, (j_1, t) is an optimal choice. \square

Proposition 2.6. Consider two tariff arcs: (i_1, j_1) and (i_2, j_2) . If $u_{si_1} \leq u_{j_2 i_1} + l_{si_2}$, we can delete arc (j_2, i_1) .

Proof. Analogous to proposition 2.5. \square

Proposition 2.7. If $u_{st} \leq l_{si_1} + l_{j_1 t}$, we can delete the tariff arc (i_1, j_1) .

Proof. The lower bound on the cost of a path going through the tariff arc (i_1, j_1) is $l_{si_1} + l_{j_1 t}$. Hence, there is no room for taxation on the tariff arc (i_1, j_1) , and thus (s, t) is always at least as good. \square

Proposition 2.8. Consider two tariff arcs: (i_1, j_1) and (i_2, j_2) . If $u_{st} \leq l_{si_1} + u_{j_1 i_2} + l_{j_2 t}$, we can delete the arc (j_1, i_2) .

Proof. The upper bound on the cost of a shortest path from s to t is u_{st} . A lower bound on the cost of a path using the tariff arc (i_1, j_1) and going to the tariff arc (i_2, j_2) is at $l_{si_1} + u_{j_1 i_2} + l_{j_2 t}$. Hence, there is no room for taxation on a path from s to t using the arc (j_1, i_2) . Thus, this arc need not be used in an optimal solution. \square

Refer to the shortest path graph model for each client as the initial shortest path graph model, and the shortest path graph model obtained after the application of the reduction methods as the reduced shortest path graph model.

Example 2.5. Consider the network depicted in Figure 2.8. This network has 14 nodes, 21 fixed cost arcs and 3 tariff arcs. The tariff arcs are given by the arcs $(1, 2)$, $(3, 4)$ and $(11, 12)$, represented by dashed arcs. The solid arcs in the network represent the fixed cost arcs. The leader on the network is dealing with one client who has a unit demand from node 13 to node 14. The leader maximizes his revenue from a linear pricing strategy.

Figure 2.9(a) shows the initial shortest path graph model for the network of Example 2.5. The solid arcs in the graph are representations of the shortest path using no tariff arcs between each node in the original network given in Figure 2.8. The cost of the arc is the cost of the corresponding shortest path in the original network between the two nodes. If no path exists between two nodes in the original network, the corresponding arc in the initial shortest path graph model is not present.

To arrive at the reduced shortest path graph model, we need to apply the reduction methods described in section 2.5.2 to the initial shortest path graph model. In Figure 2.9(a), we see we can delete the arcs $(4, 1)$, $(2, 3)$ and $(4, 11)$ of the initial shortest path graph model by applying for example Proposition 2.5 or Proposition 2.6. Furthermore, Proposition 2.7 allows us to delete the tariff arc $(11, 12)$ from the graph, since the upper bound on the cost of a path is 11 while a lower bound on the cost of a path going through the tariff arc $(11, 12)$ is 12.

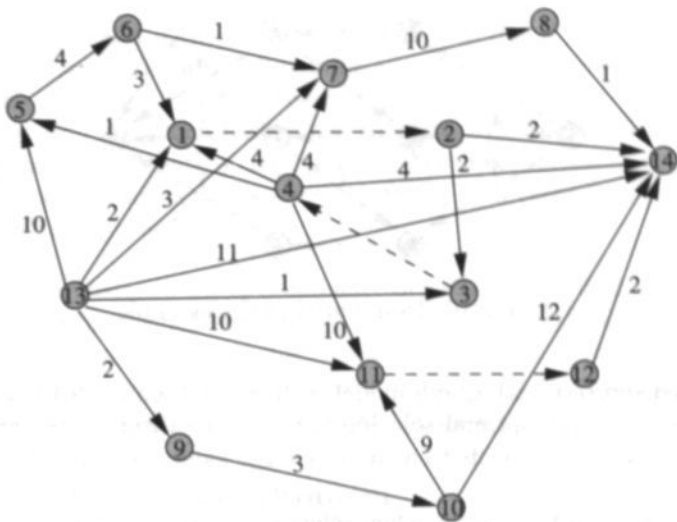


Figure 2.8: A small network with $|T| = 3$ (Example 2.5).

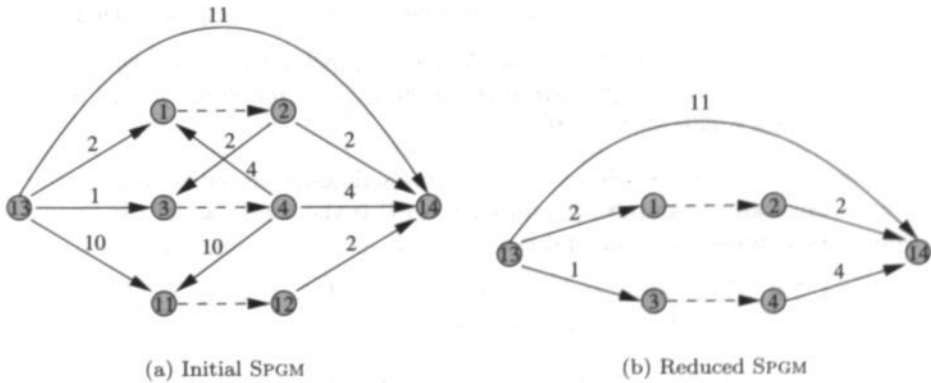


Figure 2.9: Shortest Path Graph Model for Example 2.5.

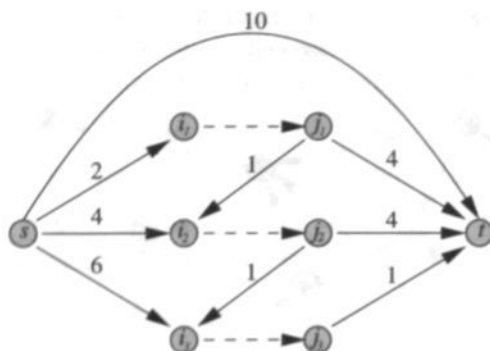


Figure 2.10: Dominated paths example.

The reduced shortest path graph model is shown in Figure 2.9(b). For this graph it is easy to see that the optimal solution is to set the tariff on the arc (1, 2) to 7, while setting all the other tariffs to a suitable large value, i.e. 11, yielding a revenue of 7 to the leader.

In the remainder of this thesis, when referring to the shortest path graph model of a client, we will refer to the reduced shortest path graph model, unless specified explicitly.

Path Reduction

By applying the propositions described in Section 2.5.2 and finding all paths in the graph, we can obtain a reduced set of paths that remains relevant for the commodity. Hopefully, this set of paths is small. It is however possible to reduce the size of this set in some instances even more by eliminating paths which are always dominated by others. The notion of path dominance is given by the following definition.

Definition 2.1. *If we can replace in all feasible solutions the path q by the path p without violating the feasibility constraints or decreasing the value of the objective function, then path p dominates path q .*

The following proposition allows us to eliminate dominated paths. Recall that T_p is the set of tariff arcs from path p , and that $\gamma_p(d)$ is the total cost of the fixed cost arcs from p for a client with demand d .

Proposition 2.9. *Consider a path p and a path q . If $T_p \subseteq T_q$ and $\gamma_q(d) \geq \gamma_p(d)$, then path p dominates path q for all tariff values.*

Proof. Suppose that path q is the shortest path taken by the client. Then

$$\gamma_q(d) + \sum_{a \in T_q} \rho_a(d) \leq \gamma_p(d) + \sum_{a \in T_p} \rho_a(d), \quad (2.20)$$

or equivalently,

$$\gamma_q(d) + \sum_{a \in T_q \setminus T_p} \rho_a(d) \leq \gamma_p(d). \quad (2.21)$$

Since $\gamma_q(d) \geq \gamma_p(d)$ and $\rho_a(d) \geq 0$ for all tariff arcs $a \in T$, inequality (2.21) only holds when $\gamma_q(d) = \gamma_p(d)$ and $\rho_a(d) = 0$ for all $a \in T_q \setminus T_p$. But then, we can replace path q by path p without violating the feasibility constraints or changing the value of the objective function since path p has the same revenue for the leader and the same cost for the client as path q . Inequality (2.21) shows furthermore that whenever $\gamma_q(d) > \gamma_p(d)$, path q can never be the path taken by a client in a feasible solution, since path p will always be cheaper. \square

Example 2.6. An instance where dominance of paths occurs is given in Figure 2.10. The tariff arcs are the dashed arcs (i_1, j_1) , (i_2, j_2) and (i_3, j_3) . The leader is dealing with a client who wants to route his demand from node s to node t . The path $q = \{s, i_1, j_1, i_2, j_2, t\}$ with fixed cost 7 is dominated by the path $p = \{s, i_1, j_1, t\}$ with fixed cost 6. The client will never select the path q .

2.5.3 River Tarification Problem

The river tarification problem is the special case of the network tarification problem where the network structure is such that any path taken by a client takes at most one tariff arc (see Section 1.3.2). The shortest path graph model for the river tarification problem has a simple structure: Obviously, for a single client, the shortest path graph model as described in Section 2.5.1 can be applied to the river tarification problem. Furthermore, since by definition, there never exists a fixed cost arc between two tariff arcs, the inner graph is the same for all commodities and all pricing strategies studied. There is thus no need to separate the shortest path graph models for each client. We can thus add arcs from each source to all tariff arcs and from all tariff arcs to each target.

One may further note that all propositions in Section 2.5.2 are redundant except for Proposition 2.7, which aims at deleting some tariff arcs. For the river tarification problem, deleting a tariff arc for a specific client without disallowing another client to take it can be achieved by deleting the arc to the tariff arc and the arc from the tariff arc for that specific client only.

Figure 2.11 gives an example of the shortest path graph model for an instance of the river tarification problem with three tariff arcs (represented by the dashed arcs) and two clients. For this example, the client (s_2, t_2) does not take the tariff arc (i_1, j_1) .

2.5.4 Insights into the Shortest Path Graph Model

The shortest path graph model focuses on the important information in a network. For example, for each client, multiple feasible paths exist in the network between two tariff arcs. To be more specific, the number of feasible paths in the network is in theory exponential for each client. However, a client is not interested in all these

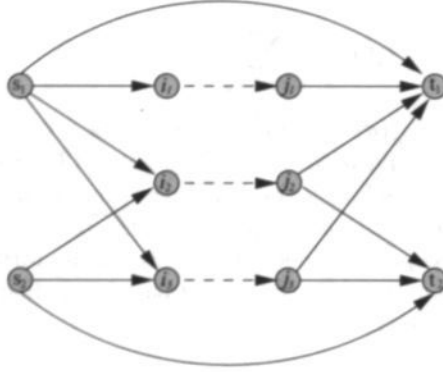


Figure 2.11: Shortest Path Graph Model for RTP.

paths. Indeed, the only relevant path between two tariff arcs is the shortest path between them.

In addition, the shortest path graph model considers only relevant tariff arcs. This is illustrated by Proposition 2.7, which deletes tariff arcs in the network that are never taken by a client. The intuition behind this proposition can be illustrated by networks with euclidean distances. See Figure 2.12, where the tariff arcs are given by the cost of the fixed cost arcs. The further away the tariff arc from the source or destination, the less likely it is used by the client. We would thus expect the shortest path graph model to perform well in case of networks where the tariff arcs are spread out evenly over the network. For each client, the number of relevant tariff arcs will be small, limiting the number of relevant paths.

Note that the size of certain instances can be *increased* by the transformation. The initial shortest path graph model is essentially a complete graph whose size depends on the number of tariff arcs. The number of fixed costs arcs in the initial shortest path graph model is of $\mathcal{O}(|T|^2)$. For a number of tariff arcs large enough, the number of fixed costs arcs in the initial shortest path graph model can be higher than in the original network. The reduction techniques can however still decrease the number of fixed cost arcs and allow for a smaller sized reduced shortest path graph model.

The number of undominated paths in a network is bounded by the number of possible subsets of T , i.e. by $2^{|T|}$: If two paths p and q have an identical set of tariff arcs, then the undominated path is the path with smallest fixed cost. It is possible to construct (artificial) networks, where the number of undominated (relevant) paths in the SPGM approaches the upper bound on the theoretical number of paths. See Example 2.7 for such a network.

Example 2.7. Consider a network with $|T|$ tariff arcs, denoted by $a_1, \dots, a_{|T|}$. A single client has unit demand from s to t . The fixed cost arcs are defined as follows. The cost of the fixed cost arcs from the source s to the tariff arc a_i , $i \in \{1, \dots, |T|\}$ is given by 3^i . The cost of the fixed cost arcs from the tariff arc a_i , $i \in \{1, \dots, |T|\}$

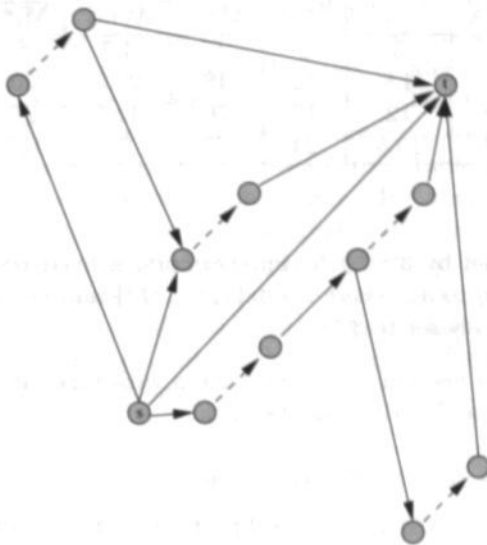


Figure 2.12: Network with euclidean distances.

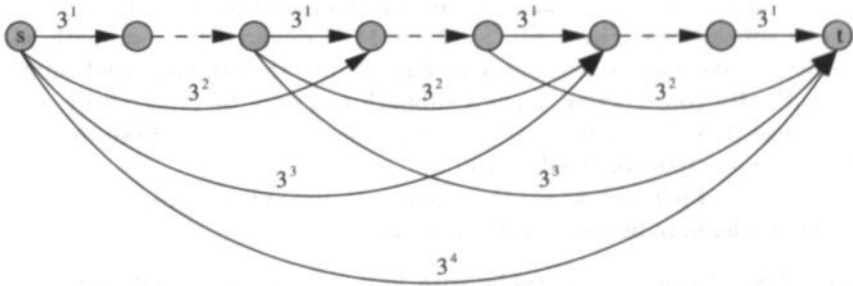


Figure 2.13: Network for Example 2.7, for $|T| = 3$.

Table 2.1: SPGM statistics for unit grid graphs.

	36 nodes, 120 arcs			64 nodes, 224 arcs			100 nodes, 360 arcs		
% $ T $	MIN	MAX	AVRG	MIN	MAX	AVRG	MIN	MAX	AVRG
2.5	2	4	3	2	6	4.3	8	18	11
5.0	3	7	4.8	12	48	22.3	20	61	40
7.5	5	19	12	17	76	42.3	27	211	132.5
10	5	40	19.8	24	115	70	126	418	268.3

to the target t is given by $3^{|T|+1-i}$. Furthermore, a fixed cost arc with cost 3^{j-i} connects a tariff arc a_i to a_j , where $i, j \in \{1, \dots, |T|\}$ and $i < j$. Finally, the cost of the fixed cost arc (s, t) is set to $3^{|T|+1}$.

As stated earlier in Section 2.5.1, any path p with tariff arcs a_i , $i \in \{1, \dots, |T_p|\}$ taken by a client can be described as the path:

$$p = \{sp_1, a_1, sp_2, a_2, \dots, sp_{|T_p|}, a_{|T_p|}, sp_{|T_p|+1}\}. \quad (2.22)$$

For the network in Example 2.7, any subpath of p of the form $\{sp_m, a_m, sp_{m+1}\}$, $m \in \{1, \dots, |T_p|\}$, has fixed cost of the form $3^i + 3^j$, where $i, j \in \{1, \dots, |T|\}$. By construction, such a subpath is bounded by a fixed cost arc with cost 3^{i+j} , as can be seen from Figure 2.13, where the network of Example 2.7 is depicted for $|T| = 3$.

For the network in Example 2.7, we show by induction that if we have a path p , then for any path q where $T_p \subseteq T_q$, $|T_q| = |T_p| + 1$, the fixed cost of p is higher than the fixed cost of q . Proposition 2.9 therefore does not allow us to eliminate dominated paths in this network. Thus, all paths in the network are undominated.

It is easy to see that this property holds for the path p using only fixed cost arcs. The cost of such a path is $3^{|T|+1}$, whereas any path with one tariff arc has a fixed cost of $3^i + 3^j < 3^{|T|+1}$, where $i + j = |T| + 1$. Now, assume we have a path p with tariff arcs T_p as described in (2.22). For any path q such that $T_p \subseteq T_q$, $|T_q| = |T_p| + 1$ exactly one subpath sp_m , $m \in \{1, \dots, |T_p| + 1\}$ is replaced by a subpath with one more tariff arc. We assume without loss of generality that the replaced subpath has fixed cost 3^k . By construction, such a subpath can only be replaced by a subpath with fixed cost $3^i + 3^j$, where $i + j = k$. Since $3^i + 3^j < 3^k$, we have that the fixed cost of p is higher than the fixed cost of q .

Another type of network where the number of undominated paths in the shortest path graph model may be large is the following.

Example 2.8. Consider grid graphs with unit costs on the fixed cost arcs in the network. We consider a client with unit demand from the upper left corner of the grid to the lower right corner. The tariff arcs are a subset of all arcs in the network, but only arcs from left to right or top to bottom.

For an $n \times n$ grid, the upper bound path using only fixed cost arcs has cost $2(n-1)$. Every tariff arc is present in the SPGM of the client (see Proposition 2.7), since the

cost of using a path with exactly one tariff arc is at most $2n - 3$. Furthermore, every time the shortest path of a client takes an additional tariff arc in the network, the cost of the path is decreased by one unit, while the path remains undominated.

Several networks using the structure of Example 2.8 were created, as follows. The number of clients in the network with unit demand was set to 4, i.e. from each corner of the grid to the opposite corner. The grids created are of size 6×6 , 8×8 and 10×10 , with two-way arcs between the nodes. The tariff arcs are assigned randomly in the grid. The other arcs in the grid are fixed cost arcs, with unit cost. The percentage of tariff arcs ranges from 2.5% to 10%. Table 2.1 shows the number of relevant paths for these networks. The column MIN, resp. MAX, indicates the minimum, resp. maximum, number of paths generated over all commodities. The column AVRG gives the average number of paths generated for each commodity for the whole data set. From Table 2.1 we can see that the number of relevant paths increases when the number of tariff arcs increases in the network. As we will see in Chapter 5 and in Chapter 6, the number of relevant paths for real life instances and random instances is considerably smaller.

Chapter 3

Complexity

The network tariffication problem was long suspected to be \mathcal{NP} -hard: Labbé et al. [41] showed that the decision version of the network tariffication problem with negative fixed costs, positive lower bounds on the tariffs and a single client is \mathcal{NP} -complete. However, the case with nonnegative fixed costs and nonnegative tariffs remained open. Marcotte, Savard, and Semet [50] showed that the network tariffication problem is a special case of the well-known travelling salesman problem (see Dantzig, Fulkerson, and Johnson [23]). Fortz and Thorup [29, 30] studied the (related) problem of optimizing the weights of the arcs in a network for a given set of projected demands so as to avoid congestion and showed that this problem is \mathcal{NP} -hard. Since imposing a sufficiently high tax on an edge in a network effectively removes it, taxation is at least as powerful as edge removal. In that context, Roughgarden [54] showed that designing networks for congestion minimization is \mathcal{NP} -hard.

Recently, Roch et al. [53] showed through a reduction from 3-SAT (see Garey and Johnson [31], Papadimitriou and Steiglitz [52]) that the network tariffication problem with a linear pricing strategy is \mathcal{NP} -hard even with a single commodity (client), nonnegative fixed costs and nonnegative tariffs. Their proof can be generalized even further to allow for the case with nonnegative fixed costs and negative tariffs.

In this chapter we give an overview of the complexity results known for the network tariffication problem. To this end, we provide in a first instance the proof given by Roch et al. [53] in their paper. We then show that the river tariffication problem remains \mathcal{NP} -hard, by presenting a reduction from 3-SAT to the network tariffication problem with multiple clients, where each client in the network takes at most one tariff arc (see Grigoriev, van Hoesel, van der Kraaij, Uetz, and Bouhtou [32]). This proof also shows that the network tariffication problem remains \mathcal{NP} -hard, even for unit demands, a fixed number of tariff values and when the leader is forced to use tariffs such that he serves (a given subset of) all clients. Consequently, the all-service network tariffication problem remains \mathcal{NP} -hard. In a second part of this chapter, polynomially solvable special cases of the network tariffication problem are discussed (see van Hoesel, van der Kraaij, Mannino, Oriolo, and Bouhtou [63]).

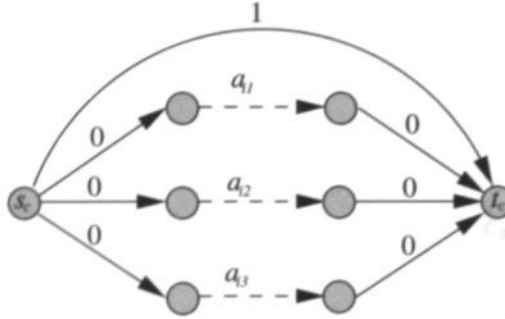


Figure 3.1: Subnetwork for $(\ell_{i1} \vee \ell_{i2} \vee \ell_{i3})$.

3.1 NP-completeness

Consider a boolean function $f : \{0,1\}^n \rightarrow \{0,1\}$ on n variables x_1, \dots, x_n , in conjunctive normal form. Such a function f is the conjunction of m clauses C_k ,

$$f = \bigwedge_{k=1}^m C_k, \quad (3.1)$$

each clause C_k being the disjunction of three literals, $C_k = (\ell_{k1} \vee \ell_{k2} \vee \ell_{k3})$. Any literal ℓ_{kj} represents either a variable x_i , or its negation \bar{x}_i , $i \in \{1, \dots, n\}$. Then f is satisfiable if there exists an assignment x_1, \dots, x_n such that at least one literal per clause is true. A function of the form (3.1) is usually called a 3-CNF function.

3.1.1 Network Tarification Problem

Roch et al. [53] showed that the network tarification problem with a linear pricing strategy is strongly NP-hard by a reduction from 3-SAT (see Garey and Johnson [31]). The proof holds for a single commodity and a linear pricing strategy on the tariff arcs.

Theorem 3.1 (Roch et al. [53]). *The network tarification problem with linear pricing strategies is strongly NP-hard.*

Proof. The 3-CNF formula f in (3.1) can be transformed in polynomial time to an instance of the network tarification problem. For each clause, create a subnetwork. This subnetwork consists of a tariff arc for each literal and two nodes s_c and t_c . A fixed cost arc with zero cost links node s_c to the source of each tariff arc and links the target of each tariff arc with the node t_c . A fixed cost arc with cost 1 furthermore exists between node s_c and t_c . See Figure 3.1 for an illustrative example.

The subnetworks corresponding to consecutive clauses c and $c+1$ are connected by a tariff arc and a fixed cost arc with a cost of 2 between node t_c and s_{c+1} . Furthermore, for every pair of tariff arcs that appear consecutively in the network and that correspond to a literal and its negation, we create a fixed cost arc with

cost 1. We refer to this fixed cost arc as the inter-clause fixed cost arc. Finally, we introduce a single client with unit demand from the source of the first subnetwork to the target of the last subnetwork. See example 3.1 for the transformation of a specific 3-CNF formula.

For the network tarification problem, if for the optimal tariffs the client in the network takes a shortest path that goes through the tariff arc a_{ij} , then the corresponding literal is true.

Refer to the shortest path from s to t using only fixed cost arcs by the path u_{st} and to the shortest path (possibly using tariff arcs) when all tariffs are set to zero by the path l_{st} . For this network, a lower bound on the cost of a path is given by the path l_{st} with total cost 0 when all tariffs are zero, while an upper bound is given by the cost of the path u_{st} with cost $m + 2(m - 1) = 3m - 2$. An upper bound on the revenue to the operator is thus given by $3m - 2$. We now show that the 3-CNF formula f is satisfiable if and only if the revenue to the leader is equal to the upper bound on the revenue of $3m - 2$ in the corresponding network.

To see this, assume the optimal revenue is equal to $3m - 2$. The leader can only achieve this revenue if the total cost of the fixed cost arcs on the shortest path taken by the client is 0. This shortest path must yield a revenue of 2 between two consecutive subnetworks and a revenue of 1 for each subnetwork. Hence, the tariff on each tariff arc between two subnetworks is equal to 2 and there exists one tariff arc with tariff 1 per subnetwork. The tariffs on the tariff arcs of each subnetwork are moreover high enough for the shortest path to not take a tariff arc with tariff 1 in two subnetworks that correspond to a literal and its negation: The corresponding inter-clause fixed cost arc would then restrict the tariff on a tariff arc connecting two subnetworks to at most 1, instead of 2. Hence, the shortest path corresponds to a consistent assignment of the variables and f is satisfiable.

Conversely, suppose that f is satisfiable and consider the path corresponding to the consistent assignment. This path does not take a variable and its negation, since it would then not correspond to a consistent assignment of the variables in f . Thus, no inter-clause fixed cost arc restricts the revenue on the tariff arcs between two subnetworks. The upper bound on the revenue of $3m - 2$ is therefore achievable on this path. \square

Example 3.1. The 3-CNF formula given by $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_4 \vee \bar{x}_2)$ can be transformed to the network given in Figure 3.2. Here, the tariff arcs a_{11}, a_{12}, a_{13} correspond to the literals x_1, x_2, x_3 and the tariff arcs a_{21}, a_{22}, a_{23} correspond to the literals $\bar{x}_1, x_4, \bar{x}_2$. The tariff arc a_1 is the tariff arc connecting the two subnetworks. The inter-clause fixed cost arcs are given by the arcs between the tariff arcs a_{11} and a_{21} and between the tariff arcs a_{12} and a_{23} . A single client has unit demand from s to t . For this example, $m = 2$. Hence, $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_4 \vee \bar{x}_2)$ is satisfiable if and only if the revenue to the operator is 4. The optimal path through the tariff arcs a_{12}, a_1 and a_{22} can achieve a revenue of 4 and corresponds with $x_2 = x_4 = 1$, with x_1 and x_3 free.

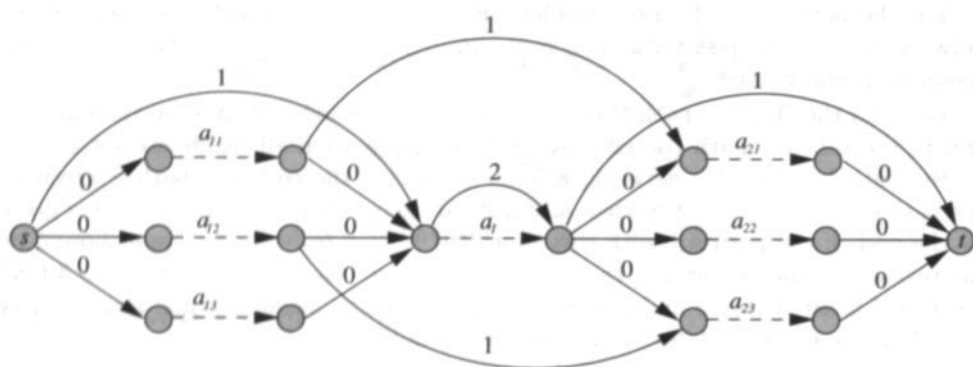


Figure 3.2: Network for $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_4 \vee \bar{x}_2)$.

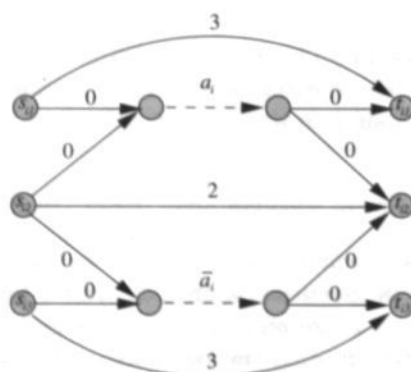


Figure 3.3: Subnetwork for variable x_i , $i \in \{1, \dots, n\}$.

3.1.2 River Tarification Problem

The reduction used by Roch et al. [53] works for network tarification problems where paths are allowed to use (and indeed, must use) many tariff arcs. To show that the river tarification problem, where we are dealing with multiple clients that take at most one tariff arc in the network, is NP-hard we need a different reduction, which we provide here (see also Grigoriev et al. [32]).

We use a reduction from 3-SAT to the river tarification problem with linear pricing strategies. Any function of the form (3.1) can be polynomially transformed to an instance of the river tarification problem as follows. For each variable x_i , $i \in \{1, \dots, n\}$, we construct a constant-size subnetwork as shown in Figure 3.3. Each of these subnetworks has three clients (commodities) with unit demand, with origin-destination pairs $\{s_{ij}, t_{ij}\}$, $j \in \{1, 2, 3\}$. Moreover, each subnetwork has two tariff arcs, a_i representing the truth assignment $x_i = 1$, and \bar{a}_i representing $x_i = 0$, as depicted in Figure 3.3.

An upper bound on the cost of routing clients 1 and 3 is given by fixed cost arcs (s_{i1}, t_{i1}) and (s_{i3}, t_{i3}) , both with cost 3. For client 2, the upper bound on the cost is given by a fixed cost arc (s_{i2}, t_{i2}) , with cost 2. The maximal revenue for each subnetwork is thus given by setting one of the tariffs to 2, and the other to 3, yielding a revenue of $2 \cdot 2 + 3 = 7$. In all other cases, the revenue is not more than 6.

Next, for each clause C_k , $k \in \{1, \dots, m\}$, we create a *clause-commodity* k with origin destination pairs $\{s_k, t_k\}$, with unit demand. Whenever a variable x_i (\bar{x}_i , respectively) appears as one of clause C_k 's literals, we connect s_k to s_{i1} (s_{i3} , respectively), and t_{i1} (t_{i3} , respectively) to t_k , using arcs of zero cost. In addition, we introduce a fixed cost arc (s_k, t_k) with cost 2, defining an upper bound of 2 for the cost of routing clause-commodity k . The so-defined instance of the river tarification problem has $2n$ tariff arcs, $3n + m$ commodities (or clients), and $7m + 11n$ fixed cost arcs, hence the transformation is indeed polynomial. Example 3.2 illustrates the transformation.

Example 3.2. The boolean function $f = (x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_4)$ on 4 variables x_1, \dots, x_4 can be transformed to the network displayed in Figure 3.4. A tariff arc a_i (resp. \bar{a}_i) represents the truth assignment $x_i = 1$ (resp. $x_i = 0$), $i \in \{1, \dots, 4\}$. \square

Theorem 3.2. *The river tarification problem with linear pricing strategies is strongly NP-hard.*

Proof. Consider the polynomial transformation defined previously. We show that a satisfying truth assignment for f exists if and only if the revenue for the river tarification problem is equal to $2m + 7n$.

Suppose there exists a satisfying truth assignment. Then at least one literal in each clause C_k , $k \in \{1, \dots, m\}$ is fulfilled. Take for each C_k one fulfilled literal and set the tariff on its corresponding tariff arc to 2. The complementary tariff arc receives tariff 3. Due to the fact that we start with a valid truth assignment, this is indeed well-defined, since either $x_i = 1$ or $x_i = 0$, but never both. This way, the total revenue from all clause-commodities is $2m$. For all remaining subnetworks, if

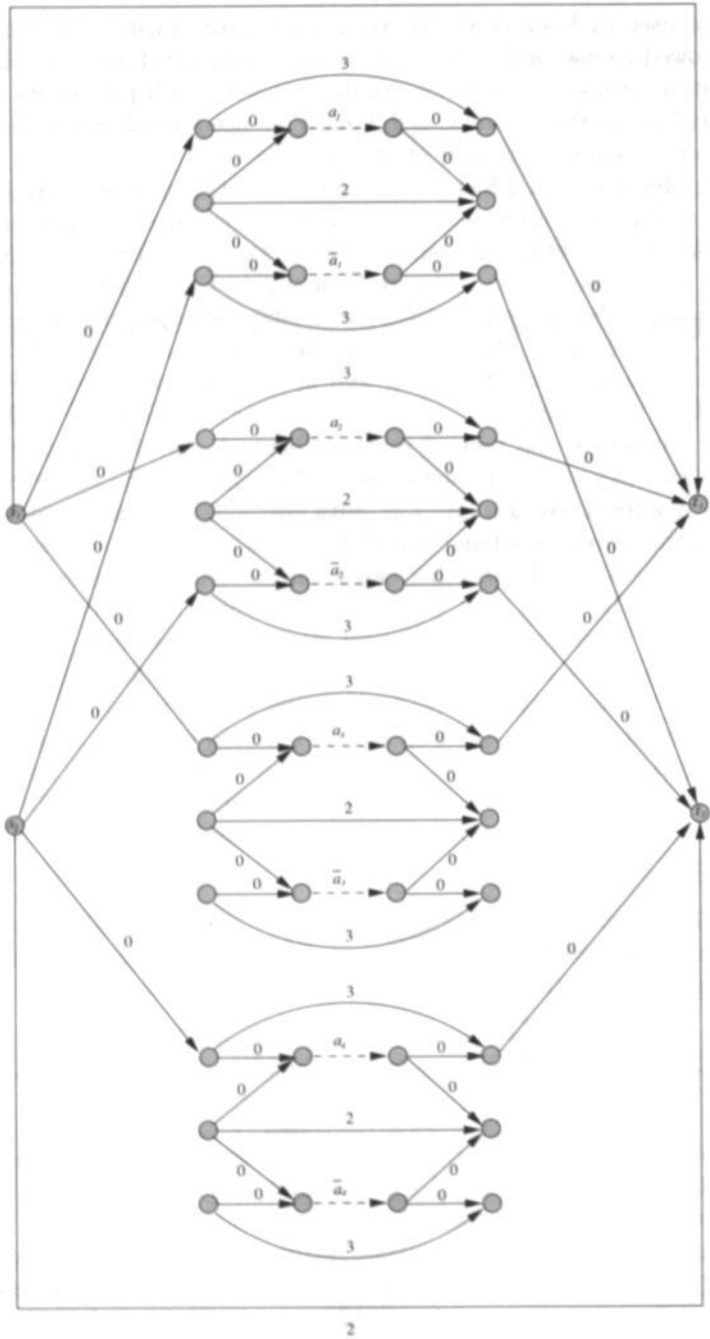


Figure 3.4: Network for $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_4)$.

any, the two tariffs can be set arbitrarily to 2 and 3, respectively. The revenue from all subnetworks is thus $7n$. Hence, the total revenue is $2m + 7n$.

Conversely, suppose there exists a set of tariffs such that the total revenue is $2m + 7n$. The maximal possible revenue created by all except the clause-commodities is $7n$, only achievable by setting one tariff arc per subnetwork to 2 and the other to 3. On the other hand, the maximal possible revenue created by the clause-commodities is $2m$. Hence, in order to achieve a revenue of exactly $2m + 7n$, we must have both, $2m$ from the clause-commodities, and $7n$ from the remaining commodities. We now define a truth assignment, setting 'true' all literals that correspond to tariff arcs of cost 2. This is a well-defined truth assignment, since we know that in each subnetwork one tariff is 2 and the other 3. Moreover, each of the m clause-commodities contributes a revenue of 2, hence it must use a path with a tariff arc of cost 2. In other words, for each clause at least one literal is 'true', hence the truth assignment satisfies all clauses. \square

3.1.3 All-service NTP

The reduction used for the proof of Theorem 3.2 shows that the river tariffication problem with linear pricing strategies remains NP-hard even for unit demands and a fixed number $k \geq 2$ of possible tariff values. The same reduction shows that the following holds:

Corollary 3.1. *The all-service network tariffication problem with linear pricing strategies is strongly NP-hard.*

Proof. See the proof of Theorem 3.2. We can use the same reduction, since the river tariffication problem is a special case of the network tariffication problem. In the above proof, we in fact show that a satisfying truth assignment for f exists if and only if the revenue for the network tariffication problem is equal to $2m + 7n$ and the leader serves all clients. \square

Remark. All pricing strategies studied (except a staircase pricing strategy) contain linear pricing as a special case. The proofs of Theorem 3.1 and Theorem 3.2 thus show that the network tariffication problem, river tariffication problem and all-service network tariffication problem are NP-hard for all these pricing strategies. For the proof of Theorem 3.1 we are dealing with a single client in the network with unit demand, while each client in the proof of Theorem 3.2 has unit demand. Therefore, there exists an equivalent staircase pricing strategy for the linear pricing strategy in each proof. NP-hardness of the network tariffication problem, river tariffication problem and all-service network tariffication problem for all pricing strategies studied thus follows from these proofs.

3.2 Polynomially Solvable Special Cases

This section focuses on polynomially solvable special cases of the network tariffication problem. Some cases (all with linear pricing) have been described in earlier work.

Labbé et al. [41] identified the case with one commodity where the path in the network taken by the client in the optimal solution is known a priori (fixed path linear tariffication problem). Furthermore they show that for multiple clients and one tariff arc (single arc linear tariffication problem) the problem is also polynomially solvable. We show that the following special cases are polynomially solvable too (see also van Hoesel et al. [63]).

For a linear pricing strategy, if the number of tariff arcs is upper bounded a priori, we show that the optimal solution can be obtained by solving a number (polynomially bounded by the number of clients) of linear programs. We refer to this type of problem as the bounded arcs linear tariffication problem. If we restrict the tariffs to be dependent on a single parameter (parametric pricing strategy, see Section 2.2), we can solve the problem in polynomial time for an unlimited number of tariff arcs. Parametric tariffication has furthermore an interesting special case, where all tariffs are restricted to be equal on all tariff arcs. The problem with multiple clients and one tariff arc for the fixed charge pricing strategy (single arc fixed charge tariffication problem) is shown to be polynomially solvable by a geometric algorithm. Moreover, we show that the fixed path linear tariffication problem can be generalized to allow for multiple clients and all different pricing strategies studied (multi-commodity fixed path tariffication problem).

Finally, we focus on the river tariffication problem. If the number of clients is bounded or if the cost on the edges of the network follows a special structure, then the river tariffication problem can also be solved in polynomial time.

3.2.1 Bounded Arcs Linear Tariffication Problem

In this section, we study the complexity of the network tariffication problem with linear pricing strategies and a bounded number of tariff arcs. The bilevel program defined in (2.16) is shown to be equivalent to a class of linear programs. Each linear program is constructed in such a way that the path taken by each client in the solution is fixed. Together, the linear programs in the class are equivalent to the linear network tariffication problem. The number of linear programs to solve is bounded by $|K|^{f(|T|)}g(|T|)$, while each linear program is of size $h(|T|)$. Note that the functions $f(|T|)$, $g(|T|)$ and $h(|T|)$ depend only on the size of the set of tariff arcs. The number of linear programs to solve is therefore polynomially bounded by the number of clients, whenever $|T|$ is upper bounded a priori. Since we can solve a linear program in polynomial time (see Khachian [39], Karmarkar [37] and Aspvall and Stone [2]), the bounded arcs linear tariffication problem is polynomially solvable.

For any client $k \in K$, consider two paths $p_1, p_2 \in P_k$. If p_1 is to be the shortest of the two paths, the constraint $\lambda_{p_1}(d_k) \leq \lambda_{p_2}(d_k)$ must hold. Thus:

$$\begin{aligned} \gamma_{p_1}(d_k) + \pi_{p_1}(d_k) &\leq \gamma_{p_2}(d_k) + \pi_{p_2}(d_k) \iff \\ \pi_{p_1}(d_k) - \pi_{p_2}(d_k) &\leq \gamma_{p_2}(d_k) - \gamma_{p_1}(d_k). \end{aligned} \quad (3.2)$$

The constraint $\lambda_{p_1}(d_k) \leq \lambda_{p_2}(d_k)$ is therefore of the form:

$$\sum_{a \in T_1} v_a - \sum_{a \in T_2} v_a \leq b^k(p_1, p_2) \quad (3.3)$$

Here, $b^k(p_1, p_2)$ is a constant and T_1 and T_2 are disjoint subsets of T . Note that T_1 contains the tariff arcs in p_1 that are not in p_2 , and T_2 contains the tariff arcs in p_2 that are not in p_1 . Tariff arcs on both paths do not appear in this constraint and hence we may assume $T_1 \cap T_2 = \emptyset$. The constant $b^k(p_1, p_2)$ is referred to in the remainder as the *switching value* for the pair (p_1, p_2) for a client $k \in K$, since p_1 is shorter than p_2 if equation (3.3) holds and p_2 is shorter than p_1 if equation (3.3) does not hold. If p_1 and p_2 have the same set of tariff arcs, then $T_1 = T_2 = \emptyset$. In that case, the left hand side of (3.3) evaluates to zero. Thus, the sign of $b^k(p_1, p_2)$ determines which of the two paths is shortest, for all possible tariff values. We compute for each client $k \in K$, for all $p_1, p_2 \in P_k$, the switching values $b^k(p_1, p_2)$.

We denote the set of all subsets of T by \mathcal{T} . The set of all pairs of mutually disjoint sets of tariff arcs is then defined as:

$$\mathcal{T}^2 := \{(T_1, T_2) \in \mathcal{T} \times \mathcal{T} : T_1 \cap T_2 = \emptyset\} \quad (3.4)$$

Let B be the increasingly ordered sequence of switching values $\{b^k(p_1, p_2) | p_1 \in P_k, p_2 \in P_k, k \in K\} \cup \{-\infty, +\infty\}$, say $B = (b_0, b_1, \dots, b_R)$. For each pair $(T_1, T_2) \in \mathcal{T}^2$, we introduce an index in the ordered sequence B , say $r \in \{1, \dots, R\}$.

For any $(T_1, T_2) \in \mathcal{T}^2$, we select an index, say $r(T_1, T_2) \in \{1, \dots, R\}$. We consider the following set of constraints:

$$b_{r(T_1, T_2)-1} \leq \sum_{a \in T_1} v_a - \sum_{a \in T_2} v_a \leq b_{r(T_1, T_2)} \quad \forall (T_1, T_2) \in \mathcal{T}^2 \quad (3.5)$$

Lemma 3.1. *Given a choice $r(T_1, T_2) \in \{1, \dots, R\}$ for all $(T_1, T_2) \in \mathcal{T}^2$, then for each client $k \in K$ there exists a path $p_k^* \in P_k$ that is the shortest path for all tariffs satisfying the constraints in (3.5), if (3.5) is not empty.*

Proof. We prove that for each client $k \in K$ and each pair of paths $p_1 \in P_k$ and $p_2 \in P_k$, one of the two is shortest for any tariffs satisfying (3.5). For two paths with the same set of tariff arcs this is trivial, as noted previously, so we can restrict ourselves to paths for which this is not the case.

For any client $k \in K$ take two arbitrary paths $p_1 \in P_k$ and $p_2 \in P_k$. As we have seen previously, the constraint $\lambda_{p_1}(d_k) \leq \lambda_{p_2}(d_k)$ is equivalent to the constraint:

$$\sum_{a \in T_1} v_a - \sum_{a \in T_2} v_a \leq b^k(p_1, p_2), \quad (3.6)$$

where $b^k(p_1, p_2)$ is one of the switching values in the sequence B . Denote the index of this switching value by \bar{r} . For the constraints in (3.5), the pair (T_1, T_2) in (3.6) has index $r(T_1, T_2)$.

If $\bar{r} \geq r(T_1, T_2)$, then

$$b_{r(T_1, T_2)-1} \leq \sum_{a \in T_1} v_a - \sum_{a \in T_2} v_a \leq b_{r(T_1, T_2)} \leq b_{\bar{r}} \quad (3.7)$$

implies that $\lambda_{p_1}(d_k) \leq \lambda_{p_2}(d_k)$. Conversely, if $\bar{r} < r(T_1, T_2)$, then $b_{\bar{r}} \leq b_{r(T_1, T_2)-1}$. Thus,

$$b_{\bar{r}} \leq b_{r(T_1, T_2)-1} \leq \sum_{a \in T_1} v_a - \sum_{a \in T_2} v_a \leq b_{r(T_1, T_2)} \quad (3.8)$$

implies that $\lambda_{p_1}(d_k) \geq \lambda_{p_2}(d_k)$. Concluding, for all tariffs satisfying the constraints in (3.5), there exists therefore a complete ordering of paths according to path length per commodity. For all $k \in K$, there is thus a path p_k^* that is the shortest for all paths in P_k . The tariff arcs on this path yield a revenue of $\pi_{p_k^*}(d_k)$. \square

Now, consider (3.5) for a choice $r(T_1, T_2) \in \{1, \dots, R\}$ for all $(T_1, T_2) \in \mathcal{T}^2$ and let the path p_k^* be the shortest for all paths in P_k for each client $k \in K$. The revenue associated with the path p_k^* is given by the linear function:

$$\pi_{p_k^*}(d_k) = \sum_{a \in T \cap p_k^*} v_a d_k \quad (3.9)$$

For a choice $r(T_1, T_2) \in \{1, \dots, R\}$ for all $(T_1, T_2) \in \mathcal{T}^2$, define the following linear program, referred to as $LP(r)$:

$$z_{LP(r)} = \begin{cases} \max & \sum_{k \in K} \sum_{a \in T \cap p_k^*} v_a d_k \\ \text{s.t.} & b_{r(T_1, T_2)-1} \leq \sum_{a \in T_1} v_a - \sum_{a \in T_2} v_a \leq b_{r(T_1, T_2)} \quad \forall (T_1, T_2) \in \mathcal{T}^2 \end{cases} \quad (3.10)$$

Refer to the optimal value of the network tariffication problem with a linear pricing strategy on the tariff arcs, defined by the bilevel program given in (2.16), as z_{MCTP} .

Theorem 3.3.

$$\max_{r(T_1, T_2)} z_{LP(r)} = z_{MCTP} \quad (3.11)$$

Proof. Given B and a choice of $r(T_1, T_2)$ for all $(T_1, T_2) \in \mathcal{T}^2$, an optimal solution (if it exists) to $LP(r)$ generates a feasible solution to z_{MCTP} . Therefore:

$$z_{LP(r)} \leq z_{MCTP} \quad (3.12)$$

Consider the optimal solution of *MCTP*. This yields a set of tariffs where for all clients $k \in K$ a path $p_k^* \in P_k$ is the shortest path. Given the optimal set of tariffs $v^* = (v_a^*)_{a \in T}$, we define $\alpha_{(T_1, T_2)}$:

$$\sum_{a \in T_1} v_a^* - \sum_{a \in T_2} v_a^* = \alpha_{(T_1, T_2)} \quad \forall (T_1, T_2) \in \mathcal{T}^2 \quad (3.13)$$

Now, for all $(T_1, T_2) \in \mathcal{T}^2$, choose $\bar{r}_{(T_1, T_2)}$ such that:

$$b_{\bar{r}_{(T_1, T_2)}-1} \leq \alpha_{(T_1, T_2)} \leq b_{\bar{r}_{(T_1, T_2)}} \quad \forall (T_1, T_2) \in \mathcal{T}^2 \quad (3.14)$$

Such an $\bar{r}_{(T_1, T_2)}$ exists for all $(T_1, T_2) \in \mathcal{T}^2$ and all clients $k \in K$ since the sequence B has values ranging from $(-\infty, +\infty)$. Given this choice of $\bar{r}_{(T_1, T_2)}$ for all $(T_1, T_2) \in \mathcal{T}^2$, the solution v^* is feasible in $LP(\bar{r})$. Thus,

$$z_{MCTP} \leq z_{LP(\bar{r})} \quad (3.15)$$

From equations (3.12) and (3.15), it follows that

$$z_{LP(\bar{r})} = z_{MCTP} \quad (3.16)$$

□

We have shown that the optimal solution of the bilevel formulation of the network tariffication problem with linear pricing strategies can be obtained by solving a number of linear problems as in (3.10). In the remainder of this section, we will compute bounds on the number of linear programs to be solved and on the size of each linear program.

Lemma 3.2. *For each client $k \in K$, the number of paths in the network is bounded by $e|T|!$.*

Proof. Given the shortest path graph model defined in Chapter 2.5, the maximum number of paths for a client $k \in K$ is defined by selecting an ordered set (possibly empty) of tariff arcs. A set of t arcs can be ordered in $t!$ ways and be chosen in $\binom{|T|}{t}$ ways. Therefore, a bound on the number of paths is given by:

$$\sum_{t=0}^{|T|} \binom{|T|}{t} t! = \sum_{t=0}^{|T|} \frac{|T|!}{(|T|-t)!} t! = |T|! \sum_{t=0}^{|T|} \frac{1}{(|T|-t)!} = |T|! \sum_{t=0}^{|T|} \frac{1}{t!} \leq e|T|! \quad (3.17)$$

□

Lemma 3.3. *The number of elements in the sequence B is bounded by $|K|(e|T|!)^2$.*

Proof. Each element of the sequence B is determined from a constraint of the form (3.3), where for a client $k \in K$ two paths $p_1 \in P_k$ and $p_2 \in P_k$ are compared. For a client $k \in K$, the number of comparisons is at most $|P_k|^2$. From lemma 3.2 we

know that $|P_k| \leq e|T|!$. Hence, the number of elements in the sequence B is at most $|K|(e|T|!)^2$. \square

Theorem 3.4. Define $f(|T|) = 4^{|T|}$, $g(|T|) = (e|T|!)^{8|T|}$ and $h(|T|) = |T|4^{|T|}$. The network tariffication problem with a linear pricing strategy can be solved by $|K|^{f(|T|)}g(|T|)h(|T|)$ linear programs, each of which is of size at most $h(|T|)$.

Proof. It follows from Theorem 3.3 that we need to solve a number of linear programs as defined in (3.10). Each of the constraints in the linear program $LP(r)$, described in (3.10), is defined by a choice of $T_1 \subseteq T$ and $T_2 \subseteq T$, such that $T_1 \cap T_2 = \emptyset$. The number of possible subsets of (T_1, T_2) is bounded by $2^{|T|}2^{|T|} = 4^{|T|}$, hence the number of constraints for each $LP(r)$ is bounded by $4^{|T|}$. For each constraint there is a choice of at most $|K|(e|T|!)^2$ right hand sides. Hence, the number of all possible linear programs $LP(r)$ is at most $\{|K|(e|T|!)^2\}^{4^{|T|}} = |K|^{f(|T|)}g(|T|)$, where $f(|T|) = 4^{|T|}$ and $g(|T|) = (e|T|!)^{8|T|}$. The size of each linear program $LP(r)$ is defined by the number of variables (at most $|T|$) and the number of constraints (at most $4^{|T|}$). Thus, the size of each linear program is bounded by $h(|T|) = |T|4^{|T|}$. \square

Note that an enumeration scheme for the single commodity linear network tariffication problem has a complexity of $\mathcal{O}(|T|4^{|T|}e|T|!)$: we can solve $e|T|!$ fixed path linear tariffication problems, which correspond to LPs of size $|T|4^{|T|}$. However, such an enumeration scheme for the multiple commodities linear network tariffication problem has to solve $(e|T|!)^{|K|}$ multi-commodity fixed path tariffication problems, which is not a polynomial function since $|K|$ is not a constant.

3.2.2 Parametric Tariffication Problem

Another polynomially solvable variant of the network tariffication problem is the parametric tariffication problem (PTP). Recall from (2.13) that for this problem we restrict the tariffs on all tariff arcs to a single parameter v : $\rho_a(d) = (\alpha_a + \beta_a v)d$ for all $a \in T$.

Given a value for the parameter v , we can calculate the shortest path tree for each client. Knowing the shortest path taken by each client for this value of v , we can determine the revenue for the leader. For each shortest path to a node $j \in N$ there are at most $|T|$ tariff arcs on the path. When increasing the tariffs on all tariff arcs, the shortest path tree will change for some tariff value v^* , decreasing by at least one tariff arc the number of tariff arcs on the shortest path to a node $j \in N$. See for example Figure 3.5, where the initial shortest path tree (Figure 3.5(a)) for an instance with $|T| = 3$ is depicted. For this example, increasing the tariff on all tariff arcs to some tariff value v^* , yields a new shortest path tree (Figure 3.5(b)) where the tariff arc (1,2) is skipped. Therefore, the number of tariff values at which the shortest path tree of a client $k \in K$ changes is $\mathcal{O}(|V|^2)$.

To keep track of the changes of the shortest path for each client, we maintain a shortest path tree. Finding the breakpoints in the shortest path tree for each client $k \in K$ corresponds to solving $|K|$ instances of the parametric shortest path problem, one for each commodity. Young, Tarjan, and Orlin [69] and Karp and Orlin [38]

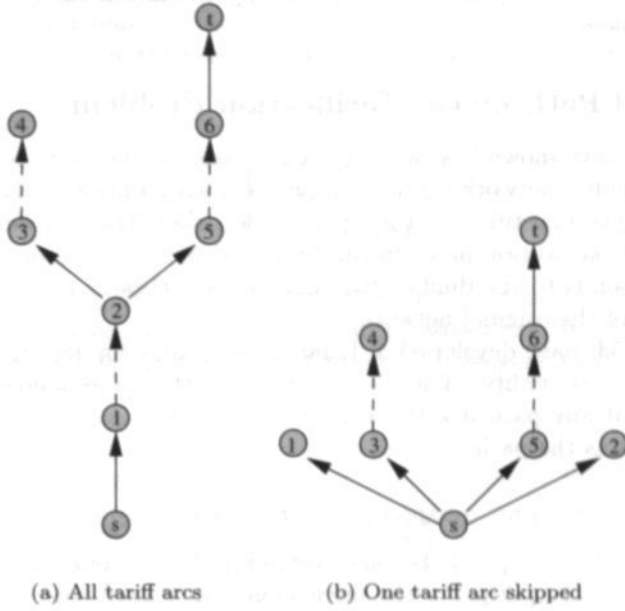


Figure 3.5: Increasing the tariff on the shortest path tree for PTP.

studied this problem and showed that the number of breakpoints is $\mathcal{O}(n^2)$. They furthermore described an algorithm which finds all these breakpoints in $\mathcal{O}(nm + n^2 \log m)$ time. Here, $|V| = n$ and $|E| = m$. With this result, we can find all breakpoints in $\mathcal{O}(|K|(nm + n^2 \log m))$. Denote these breakpoints by v_1, \dots, v_B and by $\pi_k(v_i)$ the profit to the leader from client $k \in K$ when the value of the parameter v is set to $v_i, i \in \{1, \dots, B\}$. If we denote the shortest path taken by the client $k \in K$ at parameter value v_i by p_k^* , $\pi_k(v_i)$ is given by the following:

$$\pi_k(v_i) = \sum_{a \in T \cap p_k^*} (\alpha_a + \beta_a v_i) d_k. \quad (3.18)$$

Now, let

$$i^* \in \arg \max_{i \in 1, \dots, B} \left\{ \sum_{k \in K} \pi_k(v_i) \right\}. \quad (3.19)$$

The optimal pricing strategy on each tariff arc $a \in T$ is given by:

$$\rho_a(d) = (\alpha_a + \beta_a v_{i^*}) d. \quad (3.20)$$

Note that an interesting application of parametric tariffication occurs when $\rho_a(d) = vd$ for all $a \in T$. For this particular problem, the tariffs are restricted to be equal on all tariff arcs. We refer to this pricing strategy as the uniform pricing strategy. Uniform

tarification is used in Chapter 4 to obtain an approximation algorithm for the river tarification problem.

3.2.3 Fixed Path Linear Tarification Problem

Labbé et al. [41] have shown that when the path taken by the client is known a priori, the single-commodity network tarification problem with a linear pricing strategy can be solved in polynomial time. In their paper, they show that the single-commodity fixed path linear tarification problem can be solved by a linear program. This linear program corresponds to the dual of the shortest $(s - t)$ -path problem defined on a modified graph of the original network.

Roch et al. [53] have developed a dynamic programming algorithm to solve the same problem to optimality. The idea of the algorithm is as follows. Recall from Section 2.5.1 that any path p with tariff arcs a_i , $i \in \{1, \dots, |T_p|\}$ taken by a client can be described as the path:

$$p = \{sp_1, a_1, sp_2, a_2, \dots, sp_{|T_p|}, a_{|T_p|}, sp_{|T_p|+1}\}, \quad (3.21)$$

where sp_m , $m \in \{1, \dots, |T_p| + 1\}$ is a shortest subpath using only fixed cost arcs to or from a tariff arc on the path. For the single-commodity fixed path linear tarification problem, we are given a path p^* for which we must determine the revenue maximizing tariffs v_m on the tariff arcs a_m , $m \in \{1, \dots, |T_p|\}$ such that given these tariffs, p^* is the shortest path in the network. For the dynamic programming algorithm of Roch et al. [53], we suppose without loss of generality that the nodes on the path p^* are denoted in ascending order by $\{0, \dots, 2|T_p| + 1\}$ and that we are dealing with a single client with unit demand. For each tariff arc a_m , $m \in \{1, \dots, |T_p|\}$ we furthermore denote by $h(a_m)$, respectively $t(a_m)$, the head, respectively the tail of the tariff arc. Let u_{ij} be the cost of the shortest path from node i to node j using only fixed cost arcs and let l_{ij} denote the cost of the shortest path from i to j , possibly using tariff arcs, when the tariffs are set to zero. Thus, u_{ij} is an upper bound for the cost of a path from i to j taken by the client, and l_{ij} is a lower bound for the cost of a path from i to j (see also Section 2.5.2). The revenue maximizing tariffs v_m , $m \in \{1, \dots, |T_p|\}$ are found recursively, starting with the first tariff arc on the path as follows:

$$v_m = \min_{0 \leq i < t(a_m) < h(a_m) < j \leq 2|T_p|+1} \left\{ u_{ij} - l_{ij} - \sum_{k < m} v_k \right\}, \quad (3.22)$$

The recursion in (3.22) sets the tariff on each tariff arc at its upper bound, as determined by all possible shortest paths using only fixed cost arcs from a node i to a node j on the path p^* .

3.2.4 Multi-Commodity Fixed Path Tarification Problem

We generalize the result by Labbé et al. [41] for the single-commodity fixed path tarification problem by showing that the multi-commodity fixed path tarification problem is solvable by a linear program for all pricing strategies studied.

Recall from Chapter 2 that the cost of routing a demand d on a tariff arc $a \in T$ is given by $\rho_a(d)$, whereas the cost of routing a demand on a fixed cost arc $a \in F$ is given by $\delta_a(d)$. Let b^k be the vector for each commodity $k \in K$ where for index i :

$$b_i^k = \begin{cases} -1 & \text{if } i = s_k, \\ +1 & \text{if } i = t_k, \\ 0 & \text{otherwise.} \end{cases} \quad (3.23)$$

and let the vector $y^k \in \mathbb{R}^{|A|}$ represent the flow on the arcs, for commodity $k \in K$. Define by i^+ , respectively i^- , the outgoing arcs, respectively, incoming arcs of node $i \in N$. Consider the following arc-formulation of the network tarification problem:

$$\begin{aligned} \max \quad & \sum_{k \in K} \sum_{a \in T} \rho_a(d_k) y_a^k \\ \min \quad & \sum_{k \in K} \left\{ \sum_{a \in T} \rho_a(d_k) y_a^k + \sum_{a \in F} \delta_a(d_k) y_a^k \right\} \\ \text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b_i^k \quad \forall k \in K, \forall i \in N \\ & y_a^k \geq 0 \quad \forall k \in K, \forall a \in A \end{aligned} \quad (3.24)$$

In the bilevel model (3.24), both objective functions are bilinear. Assuming that the pricing strategy is fixed, the lower level problem consists of solving $|K|$ independent shortest paths problems. Thus, as shown by Labbé et al. [41] for a linear pricing strategy, this bilevel formulation can be written as a single level problem, by replacing the lower level linear program by its optimality conditions: primal and dual feasibility, and the duality gap constraint. This yields the following (nonlinear) formulation:

$$\begin{aligned} \max \quad & \sum_{k \in K} \sum_{a \in T} \rho_a(d_k) y_a^k \\ \text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b_i^k \quad \forall k \in K, \forall i \in N \\ & \lambda_j^k - \lambda_i^k \leq \rho_a(d_k) \quad \forall k \in K, \forall a = (i, j) \in T \\ & \lambda_j^k - \lambda_i^k \leq \delta_a(d_k) \quad \forall k \in K, \forall a = (i, j) \in F \\ & \sum_{a \in T} \rho_a(d_k) y_a^k + \sum_{a \in F} \delta_a(d_k) y_a^k = \lambda_{t_k}^k - \lambda_{s_k}^k \quad \forall k \in K \\ & y_a^k \geq 0 \quad \forall k \in K, \forall a \in A \end{aligned} \quad (3.25)$$

The single-level formulation (3.25) is nonlinear due to the product $\rho_a(d_k) y_a^k$ embedded in the objective function and the duality gap constraint. However, since we assume the path taken by the clients in the optimal solution is known, the values

of y_a^k can be set to 0 or 1 a priori for all $a \in A$ and all $k \in K$. This eliminates the nonlinearity of the objective function and the duality gap constraint in (3.25). Hence, we can find the revenue maximizing tariffs for the multi-commodity fixed path tariffication problem by solving the following linear problem:

$$\begin{aligned}
 \max \quad & \sum_{k \in K} \sum_{a \in T | y_a^k = 1} \rho_a(d_k) \\
 \text{s.t.} \quad & \lambda_j^k - \lambda_i^k \leq \rho_a(d_k) & \forall k \in K, \forall a = (i, j) \in T \\
 & \lambda_j^k - \lambda_i^k \leq \delta_a(d_k) & \forall k \in K, \forall a = (i, j) \in F \\
 & \sum_{a \in T | y_a^k = 1} \rho_a(d_k) + \sum_{a \in F | y_a^k = 1} \delta_a(d_k) = \lambda_{t_k}^k - \lambda_{s_k}^k & \forall k \in K
 \end{aligned} \tag{3.26}$$

Note that when fixing the path taken by each client in the solution a priori, it is possible that no tariff values exist for which each path taken in the solution is the shortest path in the network. The linear program given in (3.26) is therefore feasible if and only if there exist tariffs such that all the fixed paths are indeed shortest paths in the network.

3.2.5 Single Arc Linear Tarification Problem

As shown by Labbé et al. [41], the network tarification problem for a linear pricing strategy is solvable by a polynomial time algorithm, in case of a single tariff arc in the network. To see this, denote that single tariff arc in the network by a^* , i.e. $T = \{a^*\}$. For each commodity, we consider two paths. Recall from Section 2.4 the shortest path u_k taken by client $k \in K$ from node s_k to node t_k , using only fixed arcs, whose cost $\gamma_{u_k}(d_k)$ is an upper bound on the cost of any path from s_k to t_k . Recall furthermore l_k , the shortest path taken by client k with smallest fixed cost from node s_k to node t_k , taken by the client when the tariff is equal to zero. Clearly, $\gamma_{l_k}(d_k)$ is a lower bound on the cost of any path from s_k to t_k . An upper bound on the revenue induced by each client $k \in K$ was therefore defined as $\mathcal{U}_k(d_k) = \gamma_{u_k}(d_k) - \gamma_{l_k}(d_k)$. When considering a unit demand for each client, this upper bound allows us to determine for each client $k \in K$ the maximum tariff for which a client $k \in K$ still takes the tariff arc a^* , denoted by $\tau_k = \mathcal{U}_k(1)$.

Let the clients $k \in K$ be ordered in nonincreasing order of the values τ_k such that:

$$\tau_1 \geq \tau_2 \geq \dots \geq \tau_{|K|}. \tag{3.27}$$

For any tariff v on the tariff arc a^* such that v is not equal to one of the values in the sequence (3.27), we can increase the tariff with $\varepsilon > 0$ and achieve a higher revenue. Thus, any optimal tariff v^* is equal to one of the values in the sequence (3.27). Moreover, for a tariff $v = \tau_i$, $i \in \{1, \dots, |K|\}$ the leader will only serve clients $k \leq i$ for which $\mathcal{U}_k(d_k) \geq v d_k$. Thus, the revenue $\Pi(\tau_i)$ for the leader at a tariff of τ_i is given by:

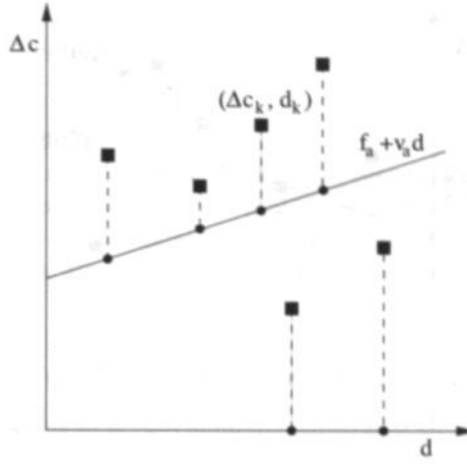


Figure 3.6: A single arc fixed charge tariffication.

$$\Pi(\tau_i) = \sum_{k \leq i} \tau_i d_k \quad (3.28)$$

The revenue maximizing tariff is thus given by τ_{i^*} , where

$$i^* = \arg \max_{i \in \{1, \dots, |K|\}} \Pi(\tau_i). \quad (3.29)$$

3.2.6 Single Arc Fixed Charge Tarification Problem

Recall the fixed charge pricing strategy on a tariff arc $a \in T$, previously defined in (2.9):

$$\rho_a(d) = \begin{cases} 0 & \text{if } d = 0 \\ f_a + v_a d & \text{if } d > 0. \end{cases} \quad (3.30)$$

The fixed charge tariffication problem can be solved in polynomial time when dealing with a single tariff arc and multiple clients, using a geometric algorithm. To this end, we denote the upper bound on the revenue $\mathcal{U}_k(d_k)$ defined in Section 2.4 for each client $k \in K$ by Δc_k . Note that we can discard clients for which Δc_k is equal to zero.

Consider Figure 3.6. In this figure, we are dealing with six clients. All clients $k \in K$ are plotted as the (square) points defined by $(d_k, \Delta c_k)$. The line in the figure represents a fixed charge tariff function $\rho_a(d)$ for the single tariff arc $a \in T$ we are considering. A commodity $k \in K$ yields revenue to the leader if and only if $f_a + v_a d_k \leq \Delta c_k$. As we can see in the figure, four clients (which are above the line) yield revenue to the leader, while two clients do not. The revenue to the leader from

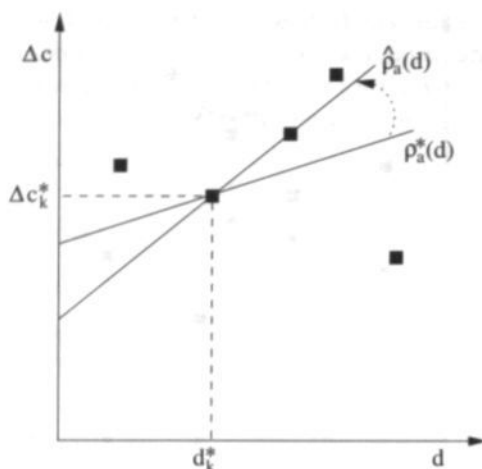


Figure 3.7: Rotating the fixed charge tariff function.

each commodity is given by the dots in the figure. Refer to the set of points $(d_k, \Delta c_k)$ for all $k \in K$ as the set U . As the following lemma shows, to find the optimal fixed charge pricing strategy, we can restrict ourselves to consider only those tariff functions that contain at least two points in U .

Lemma 3.4. *Given is a set of clients K and a single tariff arc with a fixed charge pricing strategy. The optimal fixed charge tariff function contains at least two points in U .*

Proof. Without loss of generality, we assume that the optimal fixed charge tariff function, the function $\rho_a^*(d) = f^* + v^*d$, serves the clients in the set $S \subseteq K$. Thus, for the single tariff arc $a \in T$, $f^* + v^*d_k \leq \Delta c_k$ for all clients $k \in S$. Suppose that the tariff function $\rho_a^*(d)$ contains none of the points in U . In that case, we can increase f^* , yielding a higher profit. Now, suppose the function $\rho_a^*(d)$ contains exactly one point in U , say $(d_k^*, \Delta c_k^*)$, as depicted in Figure 3.7. We will show that in that case, there exists a tariff function which contains at least two points in U , yielding at least as much revenue to the leader.

The tariff function $\rho_a^*(d)$ containing $(d_k^*, \Delta c_k^*)$ is given by the line $\rho_a^*(d) - \Delta c_k^* = v^*(d - d_k^*)$. The total revenue to the leader is thus:

$$\sum_{k \in S} [\Delta c_k^* + v^*(d_k - d_k^*)] \quad (3.31)$$

If we rotate the tariff function $\rho_a^*(d)$ around $(d_k^*, \Delta c_k^*)$ by increasing v^* by Δv , then the difference in revenue is given by $\sum_{k \in S} \Delta v(d_k - d_k^*)$. Now, if $\sum_{k \in S} (d_k - d_k^*) \geq 0$, resp. $\sum_{k \in S} (d_k - d_k^*) < 0$, then we can increase, resp. decrease, v^* until the tariff function contains at least two points in U , yielding at least as much revenue to the leader. \square

Theorem 3.5. *Given is a set of clients K and a single tariff arc with a fixed charge pricing strategy. The optimal fixed charge tariff function can be found in $\mathcal{O}(n^3)$.*

Proof. Given is the set of all clients K , where $|K| = n$, and the set of points U . From lemma 3.4 we know that the optimal fixed charge tariff function goes through at least two points in the set of points U . Since $|U| = n$, we therefore need to consider exactly $\binom{n}{2} = n(n-1)/2$ tariff functions. For each of these fixed charge tariff functions, we can calculate the revenue of the leader in $\mathcal{O}(n)$. Evaluating the revenue for the leader at all these possible lines allows us to find the optimal solution to our problem in $\mathcal{O}(n^3)$. \square

3.2.7 Bounded Number of Commodities for RTP

The river tarification problem is polynomially solvable if the number of clients $|K|$ is bounded from above by a constant. To see this, consider the following linear program, where we suppose the path taken by each client in the network is known a priori and denoted by p_k^* . Recall that P_k represents the set of all possible paths taken by a client $k \in K$.

$$\begin{aligned} \max \quad & \sum_{k \in K} \pi_{p_k^*}(d_k) \\ \text{s.t.} \quad & \lambda_p(d_k) \geq \lambda_{p_k^*}(d_k) \quad \forall k \in K, \forall p \in P_k \end{aligned} \quad (3.32)$$

The formulation in (3.32) is a linear program for all pricing strategies studied. Contrary to the network tarification problem, there are at most $|T| + 1$ paths in the network for each client. $|P_k|$ is thus bounded by $|T| + 1$. Hence, the number of constraints is polynomial in the input data. Furthermore, the number of client assignments to the tariff arcs is bounded by the number $|T|^{|K|}$ which is polynomial in the input size if the number of clients is bounded from above. Therefore, if we solve $|T|^{|K|}$ instances of (3.32), which corresponds to the multi-commodity fixed path tarification problem, we can retrieve the optimal solution. Since (3.32) is a linear program, the optimal solution can be retrieved in polynomial time.

3.2.8 Special Cost Structure for RTP

For several special cost structures of the network, the river tarification problem is polynomially solvable. These cost structures remind of the special combinatorial structures of the distance matrix for the Travelling Salesman Problem (see Burkard, Deineko, van Dal, van der Veen, and Woeginger [16], Lawler, Lenstra, Rinnooy-Kan, and Shmoys [43]).

Denote for each commodity $k \in K$ the upper bound on the cost of the path from s_k to t_k by u_k , $k \in \{1, \dots, |K|\}$. For each node $n \in N$ define a constant $c_n \geq 0$. The cost of each fixed cost arc (i, j) is given by $(c_i + c_j)$, except for the arcs (s_k, t_k) which represent the upper bounds u_k . The source and target of each commodity is furthermore connected to all tariff arcs. See Figure 3.8 for an example. In Figure 3.8, the cost of getting to the tariff arc 1 from node s_1 is given by $(c_1 + c_3)$. We can now transform the costs of fixed cost arcs such that the cost of using a tariff arc is the

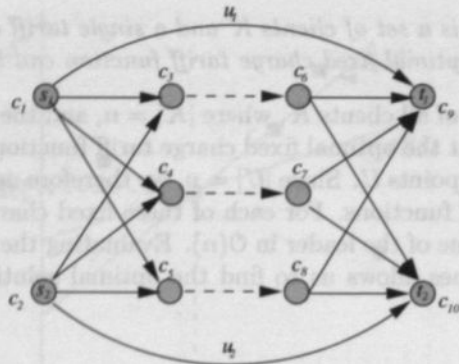


Figure 3.8: Special cost structure of network ($|K| = 2, |T| = 3$).

same for all commodities, by incorporating the costs c_{s_k} and c_{t_k} in the upper bound: $u_k \leftarrow u_k - (c_{s_k} + c_{t_k})$. For this new instance, in an optimal solution all clients use the same tariff arc. We can thus find the optimal solution of this special case of the river tarification problem by solving $|T|$ instances of the single arc linear tarification problem.

Chapter 4

Approximation

The first approximation result for the network tariffication problem was presented by Roch et al. [53] who showed that the network tariffication problem with a single client and a linear pricing strategy is approximable with $\mathcal{O}(\log(|T|))$. In Section 4.1 we propose a $|T|$ -approximation algorithm for the case with multiple clients, where each client takes at most one tariff arc in the network (river tariffication problem). We show in Section 4.2 that the all-service network tariffication problem presented in Section 1.3.3 is inapproximable with $\mathcal{O}(|T|^{1-\epsilon})$, unless $\mathcal{ZPP} = \mathcal{NP}$. See also Grigoriev et al. [32]. In Section 4.3 we present some worst-case examples for several algorithms based on the polynomial special cases of the network tariffication problem described in Chapter 3. For all results in this chapter we only consider a linear pricing strategy.

4.1 River Tariffication Problem

We show that, in general, a uniform pricing strategy (uniform tariffication) provides a $|T|$ -approximation for the river tariffication problem with linear pricing strategies. Whenever demands and costs on the arcs are polynomially bounded (in terms of $|T|$), we improve this to an $\mathcal{O}(\log(|T|))$ -approximation.

We assume the network has a special structure, as described in Section 2.5.3. Given the structure of the shortest path graph model, we may assume without loss of generality that all fixed cost arcs incident with the target nodes have zero cost (by adding their respective costs to the fixed cost arcs incident with the source node). Moreover, since any path taken by a client involves exactly one fixed cost arc with non-zero cost, we may assume without loss of generality that the costs of the fixed cost arcs $a \in F$ are integral (by scaling, if necessary).

Due to the integrality of the costs of the fixed cost arcs, it is immediate that any reasonable solution for the river tariffication problem will adopt only tariffs which are integral, too. Notice, however, that this is not true for the network tariffication problem, because a path chosen by a client can consist of several tariff arcs. There, even if the costs of the fixed cost arcs are integral, it may be that an optimal solution utilizes fractional tariffs (see Example 4.1).

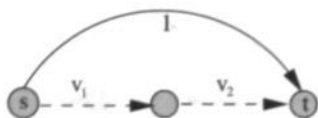


Figure 4.1: Optimal fractional tariffs exist for the NTP.

Example 4.1. See the network of Figure 4.1. For this network, we are dealing with a single client from s to t , with unit demand. The tariff arcs are given by the dashed arcs. One of the optimal solutions corresponds to setting the tariffs to $v_1 = \frac{1}{2}, v_2 = \frac{1}{2}$.

For convenience, the pricing strategy on the arcs is denoted by $\rho_a(d) = v_a d$, for all tariff arcs $a \in T$ and by $\delta_a(d) = c_a d$, for all fixed cost arcs $a \in F$.

4.1.1 $|T|$ -approximation Algorithm

We consider the uniform tariffication problem (UTP), described in Section 2.2. Recall that uniform tariffication is the special case of the network tariffication problem, with the additional restriction that all tariffs are required to be identical. The pricing strategy on each tariff arc $a \in T$ is therefore defined as:

$$\rho_a(d) = vd, \quad (4.1)$$

where $v \geq 0$ is identical for all tariff arcs. As shown in Section 3.2.2, the uniform tariffication problem can be solved in polynomial time, even in the general setting where clients may use paths with several tariff arcs.

We are interested in the loss that can be experienced by adopting a uniform tariffication policy. Therefore, denote by Π^{UTP} the revenue for an optimal uniform tariffication, and by Π^{OPT} the revenue for an optimal (not necessarily uniform) tariffication. By definition, $\Pi^{UTP} \leq \Pi^{OPT}$.

Lemma 4.1. *If an optimal tariffication for the river tariffication problem with revenue Π^{OPT} utilizes at most r different tariffs, then for the optimal uniform tariffication, $\Pi^{UTP} \geq \Pi^{OPT}/r$.*

The proof of this lemma is indeed trivial once the structure of an optimal solution to the river tariffication problem is clear. To this end, consider an optimal non-uniform tariffication with tariffs $v_1 \leq \dots \leq v_{|T|}$, and let D_i be the total demand on an arc a_i with tariff v_i , $i \in \{1, \dots, |T|\}$. By $D = \sum_{i=1}^{|T|} D_i$ we denote the total demand served by the operator. Then the revenue created by this solution is the area under the following ‘staircase’ function $f : [0, D] \rightarrow [0, \infty)$, illustrated in Figure 4.2.

$$f(x) = v_i \quad \text{for all } x \text{ with } \sum_{j < i} D_j \leq x < \sum_{j \leq i} D_j, \quad i \in \{1, \dots, |T|\}. \quad (4.2)$$

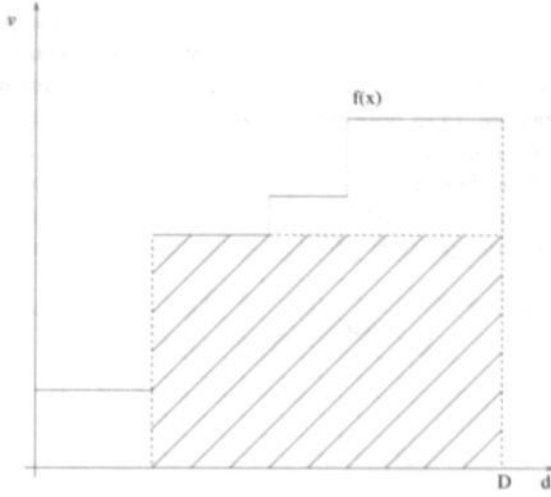


Figure 4.2: Staircase function $f(x)$ with inscribed rectangle.

Proof of Lemma 4.1. Consider any of the rectangles inscribed under the graph of function $f(x)$, with area $T_i := v_i \cdot \sum_{j \geq i} D_j$, as depicted in Figure 4.2. Then it holds that $\Pi^{UTP} \geq T_i$ for all $i \in \{1, \dots, |T|\}$, since the area of any such rectangle is a lower bound for the revenue yielded by the optimal uniform tariff Π^{UTP} . This claim is true because, starting with the optimal set of tariffs $v_1, \dots, v_{|T|}$, and levelling all tariffs to some value v_j , creates a revenue loss of at most $\sum_{i: v_i < v_j} D_i v_i + \sum_{i: v_i > v_j} D_i (v_i - v_j)$. (Notice that this does not hold for the network tariffication problem.) Hence, if only r different tariffs are utilized, we consider the r (inclusion-)maximal rectangles under the function f , say T_{i_1}, \dots, T_{i_r} , and get $r \cdot \Pi^{UTP} \geq \sum_{j=1}^r T_{i_j} \geq \Pi^{OPT}$. \square

Since $r \leq |T|$, Lemma 4.1 yields the following theorem. Tightness of the result will be shown below in Section 4.1.3, using Example 4.2.

Theorem 4.1. *Uniform tariffication is a $|T|$ -approximation for the river tariffication problem, and this bound is tight.*

4.1.2 $\mathcal{O}(\log(|T|))$ -approximation Algorithm for Bounded Input Data

We next derive an improved bound on the quality of uniform tariffication policies for the case of bounded input data. To this end, denote by v_{\max} an upper bound on the maximal possible tariff. For example, a trivial value for v_{\max} is given by the maximal cost of any fixed cost arc. Since we are dealing with a linear pricing strategy on all arcs, $v_{\max} = \max_{a \in F} c_a$. Then, if D is the total demand captured by the operator in an optimal non-uniform tariffication,

$$U := D \cdot v_{\max} \quad (4.3)$$

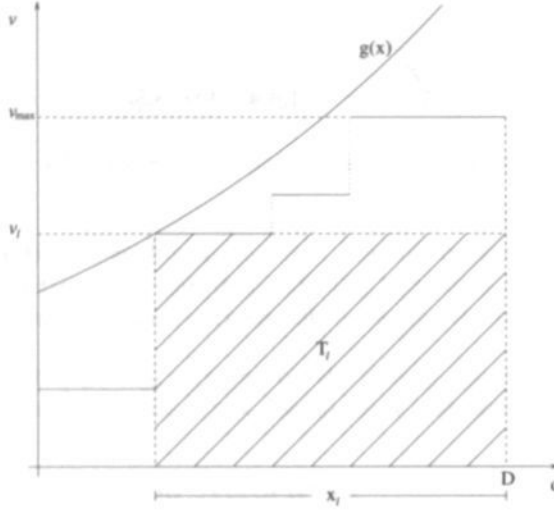


Figure 4.3: Illustration for the proof of Lemma 4.2.

is a trivial upper bound for the optimal revenue Π^{OPT} for non-uniform tariffication.

Lemma 4.2. *If Π^{OPT} is the revenue for an optimal tariffication for the river tariffication problem, and U is the upper bound on Π^{OPT} as defined in (4.3), then for the optimal uniform tariffication, $\Pi^{UTP} \geq \Pi^{OPT}/(1 + \log U)^1$.*

Proof. Indeed, we will even prove a slightly stronger result than claimed. First recall the definition of the staircase function f in (4.2), as well as the inscribed rectangles, with areas $T_i = v_i \cdot \sum_{j \geq i} D_j$. Let ℓ be the index of the maximal of these $|T|$ rectangles, with area T_ℓ . Let $x_\ell := \sum_{j \geq \ell} D_j = T_\ell/v_\ell$. We show

$$\Pi^{UTP} \geq \frac{\Pi^{OPT}}{1 + \log(U/T_\ell)}. \quad (4.4)$$

Lemma 4.2 then follows, because we know that the optimal tariff only adopts integral values, and $T_\ell \geq 1$. To prove (4.4), we define the function

$$g(x) := \frac{T_\ell}{D - x} \text{ for } x \in [0, D). \quad (4.5)$$

We claim that $g(x) \geq f(x)$ for $x \in [0, D)$. To see this, take any x with $\sum_{j < i} D_j \leq x < \sum_{j \leq i} D_j$, then $f(x) = v_i$ by definition. However,

$$g(x) = \frac{T_\ell}{D - x} \geq \frac{T_\ell}{D - \sum_{j < i} D_j} = \frac{T_\ell}{\sum_{j \geq i} D_j} = \frac{T_\ell}{T_i/v_i} \geq v_i = f(x), \quad (4.6)$$

¹With $\log x$ we denote the natural logarithm, i.e., $e^{\log x} = x$.

where the first inequality follows by choice of x , and the last follows by choice of ℓ as the index of the largest rectangle.

Hence, the area under the staircase function, which equals Π^{OPT} , can be upper bounded in terms of the area defined by the function $g(x)$, as depicted in Figure 4.3. To compute this area, we partition it into three parts, namely the rectangle T_ℓ itself, the area under $g(x)$ on the domain $x \in [0, D - x_\ell]$, as well as the area to the right of $g(x)$ on the domain $v \in [v_\ell, v_{\max}]$. The latter is the integral of the function $D - g^{-1}(v) = T_\ell/v$ on the domain $[v_\ell, v_{\max}]$. We thus obtain the following.

$$\Pi^{OPT} \leq T_\ell + \int_0^{D-x_\ell} \frac{T_\ell}{D-x} dx + \int_{v_\ell}^{v_{\max}} \frac{T_\ell}{v} dv, \quad (4.7)$$

thus

$$\begin{aligned} \Pi^{OPT} &\leq T_\ell [1 + \log D + \log v_{\max} - \log v_\ell - \log x_\ell] \\ &\leq T_\ell [1 + \log U - \log T_\ell] \\ &\leq T_\ell [1 + \log(U/T_\ell)], \end{aligned} \quad (4.8)$$

and since $T_\ell \leq \Pi^{UTP}$, claim (4.4) follows. \square

Notice that claim (4.4) confirms the following geometric intuition: The closer the staircase function $f(x)$ is to the straight line $x \mapsto (v_{\max}/D) \cdot x$, the closer T_ℓ is to $U/4$, which yields an approximation ratio of $(1 + \log 4) \approx 2.4$ for uniform tarification. Geometric intuition suggests a ratio of roughly 2, and the additional 0.4 is due to our analysis, caused by the difference between the functions $g(x)$ and $f(x)$. In Section 4.1.4, we compare the quality of uniform versus non-uniform tarification for the river tarification problem, based on numerical results for instances obtained from France Télécom.

Theorem 4.2. *If both demand per client d_k , $k \in \{1, \dots, |K|\}$, and costs c_a on the arcs $a \in F$ are polynomially bounded in terms of $|T|$ (the number of tariff arcs), then uniform tarification is an $\mathcal{O}(\log(|T|))$ -approximation for the river tarification problem.*

Proof. Since $d_k \in \text{poly}(|T|)$ for all k and $c_a \in \text{poly}(|T|)$ for all $a \in F$, we have that $v_{\max} \in \text{poly}(|T|)$. Therefore, $1 + \log U \in \mathcal{O}(\log(|T|))$. Thus, from Lemma 4.2 the theorem follows. \square

4.1.3 Tightness of the Approximation

Finally, let us show tightness of the bounds in Lemma 4.1 and Lemma 4.2, and thus also tightness of Theorem 4.1.

Example 4.2. Given $|T|$ commodities and $|T|$ tariff arcs. Every commodity is operating its own subnetwork with just one tariff arc, that is, the entire network consists of $|T|$ disjoint subnetworks and each of them contains one commodity and

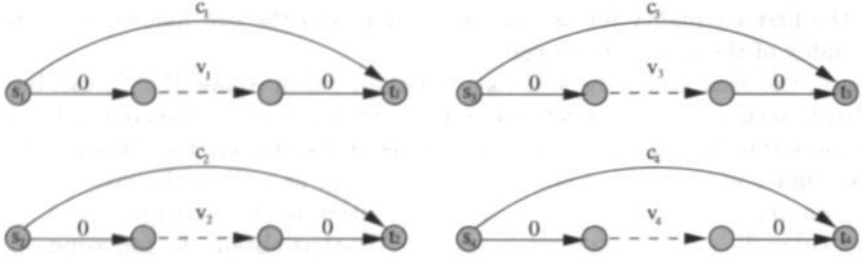


Figure 4.4: The analysis of uniform tariffication policies is tight.

one tariff arc. See also Figure 4.4. Fix an integer $b \geq 2$ and let the demand in subnetwork k be given by $d_k = b^k - b^{k-1}$, $k \in \{1, \dots, |T|\}$. This way, the total demand D equals $b^{|T|} - 1$. Moreover, the maximal revenue for subnetwork k is limited by a fixed cost arc (s_k, t_k) , with cost $c_k = b^{2|T|-k}$. Hence, the maximal tariff v_{\max} equals $b^{2|T|-1}$.

In the optimal solution, the tariff for each subnetwork k is set to its maximal value, $b^{2|T|-k}$. Each subnetwork therefore contributes a revenue of $b^{2|T|} - b^{2|T|-1}$, and $\Pi^{OPT} = |T|(b^{2|T|} - b^{2|T|-1})$. The optimal uniform tariffication consists in setting the tariff on all tariff arcs to $b^{|T|}$. This way, every unit of demand creates a profit of $b^{|T|}$, yielding a total revenue of $b^{2|T|} - b^{|T|}$. Other (reasonable) uniform tariffs would be values $b^{2|T|-k}$, $k \in \{1, \dots, |T| - 1\}$. When setting the uniform tariff to $b^{2|T|-k}$, only clients $l \leq k$ are served. The revenue from these clients is equal to:

$$b^{2|T|-k} \left[\sum_{l=1}^k b^l - b^{l-1} \right] = b^{2|T|-k} [b^k - 1]. \quad (4.9)$$

This yields a total revenue of $b^{2|T|} - b^{2|T|-k}$, which is less. Therefore, we obtain

$$\Pi^{UTP} / \Pi^{OPT} = \frac{b^{2|T|} - b^{|T|}}{|T|(b^{2|T|} - b^{2|T|-1})} \leq \frac{b^{2|T|}}{|T|(b^{2|T|} - b^{2|T|-1})} = \frac{1}{|T|} \cdot \frac{b}{b-1}. \quad (4.10)$$

Now, observe that in the optimal solution $|T|$ different tariffs are utilized. Lemma 4.1 (Theorem 4.1, respectively) suggests that uniform tariffication provides a $|T|$ -approximation. Example 4.2 proves that this is best possible, since b can be chosen arbitrarily large.

Moreover, Lemma 4.2, respectively claim (4.4), suggests that uniform tariffication is a $(1 + \log(U/T_\ell))$ -approximation. Here, T_ℓ is the largest rectangle under the function f ; see the proof of Lemma 4.2. In Example 4.2, we have $U = v_{\max} \cdot D = (b^{|T|} - 1) \cdot b^{2|T|-1}$ and $T_\ell = b^{2|T|}$. We thus have $U/T_\ell = (b^{|T|} - 1)/b$. Hence, $(1 + \log(U/T_\ell)) = 1 + \log(b^{|T|} - 1) - \log b \leq 1 + (|T| - 1) \log b \leq |T| \log b$, for $b \geq 3$. Hence, for $b \geq 3$, Lemma 4.2 yields that uniform tariffication is a $|T|$ -approximation on this example (the hidden constant being $\log b$), while the same Example 4.2 shows

Table 4.1: Quality of uniform tarification on France Télécom instances.

Data	N	A	T	K	Π^{OPT}	Π^{UTP}	%
RTN1	29	94	7	15	841	624	74%
RTN2	29	98	6	21	4099	3496	85%
RTN3	59	206	10	13	1118	880	79%
RTN4	59	204	10	20	2217	1512	68%
RTN5	49	120	9	21	74948	55968	74%
RTN6	33	116	15	12	28166	20328	72%
Average:							75%

that this is indeed best possible. Notice that this does not contradict the claim of Theorem 4.2, since in Example 4.2 neither tariffs nor demands are polynomially bounded in $|T|$.

Notice furthermore that the proofs we presented do not carry over straightforwardly to the network tarification problem. The reason is that we made use of the fact that any unit of demand utilizes at most one tariff arc. This is no longer true in general.

4.1.4 Numerical Results

As stated previously, geometric intuition suggests a worst-case ratio for uniform tarification of approximately 2. In this section, we present numerical results based on several instances of the river tarification problem, obtained from France Télécom. These instances represent telecommunication networks for the international interconnections market, one of the applications described in Section 1.2.

We compare the optimal solution using uniform tariffs (Π^{UTP}) to the solution using non-uniform tariffs (Π^{OPT}). The value of the optimal non-uniform tarification solution is calculated using the model and mixed integer programming formulation described later on in Chapter 5. The value of Π^{UTP} is calculated using the same formulation, with the extra restriction that all tariffs should be equal. As such, we do not compare the actual running times of both methods, but are more interested in how efficient uniform tarification is for these instances.

Table 4.1 gives a description of each network, stating the number of nodes, arcs, tariff arcs and clients. The value of the optimal solution found is given in the columns with header Π^{OPT} and Π^{UTP} . The final column gives the approximation ratio as a percentage of the optimal non-uniform solution. These results show that, for these instances, uniform tarification yields on average 75% of the optimal non-uniform tarification.

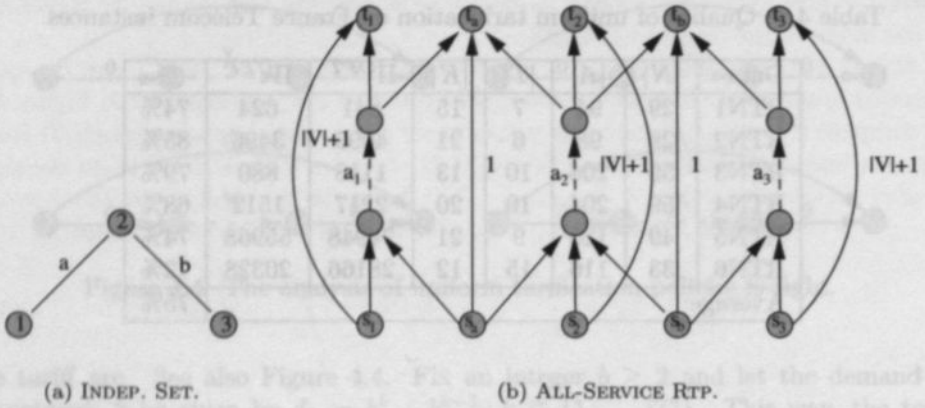


Figure 4.5: Reduction of INDEPENDENT SET to ALL-SERVICE RTP.

4.2 Inapproximability of the All-service NTP

In this section, we consider the variation of the network tariffication problem where the leader must capture the demand of all clients, i.e. service all clients in the network. The all-service network tariffication problem was described in Section 1.3.3. \mathcal{NP} -hardness of this problem follows by our previous reduction presented in Section 3.1.2. We show:

Theorem 4.3. *For any $\varepsilon > 0$, the existence of a polynomial time approximation algorithm for the all-service network tariffication problem with $|K|$ clients and $|T|$ tariff arcs*

- *with worst case ratio $\mathcal{O}(|T|^{1/2-\varepsilon})$ implies $\mathcal{P} = \mathcal{NP}$;*
- *with worst case ratio $\mathcal{O}(|T|^{1-\varepsilon})$ implies $\mathcal{ZPP} = \mathcal{NP}$;*
- *with worst case ratio $\mathcal{O}(|K|^{1/4-\varepsilon})$ implies $\mathcal{P} = \mathcal{NP}$;*
- *with worst case ratio $\mathcal{O}(|K|^{1/2-\varepsilon})$ implies $\mathcal{ZPP} = \mathcal{NP}$.*

Proof. We present an approximation preserving reduction from INDEPENDENT SET (see Garey and Johnson [31]) to the all-service river tariffication problem. Since the river tariffication problem is a special case of the network tariffication problem, this proves the theorem.

Assume we are given a graph $G = (V, E)$, and the problem is to find a maximum cardinality subset $V' \subseteq V$ of vertices such that no two vertices in V' are connected by an edge. The transformation works as follows. For every vertex $v \in V$ we introduce a client with origin-destination pair $\{s_v, t_v\}$ and demand $d_v = |E|$, and a corresponding tariff arc a_v . We connect the source s_v to the tail of the tariff arc a_v , and the head

of a_v to the destination t_v , using zero cost fixed cost arcs. Moreover, there is a fixed cost arc (s_v, t_v) with cost $(|V| + 1)$ for all vertices $v \in V$. For every edge $e \in E$ we introduce a client with origin-destination pair $\{s_e, t_e\}$ and unit demand. The upper bound on the cost of routing this demand is given by the fixed cost arc (s_e, t_e) with cost 1. For all edges $e \in E$ and all vertices $v \in V$ with $v \in e$, we furthermore introduce fixed cost arcs $(s_e, \text{tail}(a_v))$ and $(\text{head}(a_v), t_e)$, with zero cost. This transformation results in an instance of the all-service river tarification problem with $|V|$ tariff arcs, and $|V| + |E|$ clients. Figure 4.5 gives an example of such a transformation for a graph $G = (V, E)$ with 3 nodes and 2 edges.

We claim that G has an independent set of cardinality at least k if and only if there exists a tariff policy for the all-service river tarification problem with a total revenue of $|V||E|(k + 1) + |E|$.

First, assume that G has an independent set V' of cardinality k . For all $v \in V'$, set the tariff on the corresponding tariff arc a_v to $|V| + 1$, and all other tariffs to 1. By the definition of an independent set, for any edge $e = (v, u) \in E$ at least one of the vertices, v or u , is not in V' . Therefore, the tariff of at least one of the tariff arcs, a_v or a_u is 1. All clients corresponding to an edge e can thus be served, using one of the tariff arcs a_v or a_u . The clients (s_v, t_v) corresponding to the vertices $v \in V$ are also served, since the upper bound of $|V| + 1$ is not exceeded with the so-defined tariffs. Hence, all clients are served. The revenue consists of $|E|$ from all clients corresponding to the edges E of G , $|E|(|V| + 1)k$ from the clients corresponding to the independent set V' , and $|E|(|V| - k)$ from the clients corresponding to $V \setminus V'$. That yields a total revenue of $|E||V|(k + 1) + |E|$.

Conversely, assume that there exists a set of tariffs that captures all demands, such that the revenue is $|E||V|(k + 1) + |E|$. We will show that this implies that the graph G has an independent set of cardinality at least k . Since all demands are captured at this tarification strategy, for any edge $e = (v, u) \in E$, the tariff on at least one of the arcs, a_v or a_u , is 1. Consider the set of vertices $V' := \{v \in V : t_{a_v} > 1\}$. By definition, no pair of nodes $v, u \in V'$ is connected by an edge. Hence, V' is an independent set in G . Let $k' := |V'|$. The revenue is equal to $|E| + |E|(|V| - k') + |E|(|V| + 1)k' = |E||V|(k' + 1) + |E|$, which by assumption is at least as large as $|E||V|(k + 1) + |E|$. This implies that $k' \geq k$ and thus that V' is an independent set in G of cardinality $k' \geq k$.

Now, let us assume that we have an α -approximation algorithm \mathcal{A} for the all-service river tarification problem, with $\alpha \geq 1$. Consider any instance $G = (V, E)$ of INDEPENDENT SET, and the all-service river tarification problem resulting from the above reduction. We can assume that both the optimal solution and the solution produced by \mathcal{A} only utilize tariff values 1 or $|V| + 1$, because any tariff greater than 1 and not equal to $|V| + 1$ can be turned into $|V| + 1$ with a revenue gain. So $\Pi^{OPT} = |E||V|(k + 1) + |E|$ for some k , and $\Pi^{\mathcal{A}} = |E||V|(k' + 1) + |E|$ for some k' . The first part of the proof yields that the maximal independent set of G has size k , and algorithm \mathcal{A} can be used to find an independent set of size at least k' . Moreover,

$$\frac{1}{\alpha} \leq \frac{|E||V|(k' + 1) + |E|}{|E||V|(k + 1) + |E|} = \frac{1 + \frac{1}{|V|} + k'}{1 + \frac{1}{|V|} + k} \leq \frac{2 + k'}{1 + k}, \quad (4.11)$$

hence $k' \geq (k+1)/\alpha - 2$. In other words, we have an $\mathcal{O}(\alpha)$ -approximation algorithm for the INDEPENDENT SET problem.

It is now well known from work of Håstad [34] that the CLIQUE (see Garey and Johnson [31]) problem in a graph $G = (V, E)$, and hence also the INDEPENDENT SET problem cannot have a polynomial time approximation algorithm with worst case guarantee $\mathcal{O}(|V|^{1/2-\epsilon})$ unless $\mathcal{P} = \mathcal{NP}$, and that it cannot have a polynomial time approximation algorithm with worst case guarantee $\mathcal{O}(|V|^{1-\epsilon})$ unless $\mathcal{ZPP} = \mathcal{NP}$. Since the number of tariff arcs $|T|$ in our transformation equals $|V|$, the first two claims of the theorem follow. Since the number of clients $|K|$ in our transformation equals $|V| + |E| \in \mathcal{O}(|V|^2)$, the second two claims follow. For a description of the class \mathcal{ZPP} , see Motwani and Raghavan [51]. \square

4.3 Failing Approximation Algorithms for the NTP

Consider the polynomially solvable special cases described in Section 3.2. These special cases can prove quite useful for approximation algorithms, as shown in Section 4.1.1 for uniform tariffication. Another example where a polynomially solvable special case is successfully used in an approximation algorithm is in the paper by Roch et al. [53]. They use the fixed path linear tariffication problem to obtain their $\mathcal{O}(\log(|T|))$ -approximation algorithm of the single client network tariffication problem.

In this section, we present worst-case examples for several algorithms which use the previously mentioned polynomially solvable special cases. These algorithms were developed while attempting to design an approximation algorithm for the network tariffication problem or the river tariffication problem. The worst-case examples show why they cannot serve this purpose.

From Section 3.2.1, we know that solving the network tariffication problem with linear pricing strategies is polynomially bounded by the number of clients, whenever $|T|$ is upper bounded a priori. Consider therefore the k -tariff arcs algorithm, where we solve the network tariffication problem with linear pricing strategies for at most $k \leq |T|$ tariff arcs. For this algorithm, the client thus takes at most k tariff arcs in the network.

We define a network showing that the k -tariff arcs algorithm can be arbitrarily far from the optimal solution. For this network, we are dealing with a single client with unit demand from node s to node t . The tariff arcs in T are the arcs a_i , $i = \{1, \dots, |T|\}$, while the fixed cost arcs are defined as follows: for all tariff arcs a_i , $i = \{1, \dots, |T|\}$, the cost of the fixed cost arcs from the source node s to the tariff arc a_i is set to $(i-1)$, while the cost of the fixed cost arcs from the tariff arc a_i to the target node t is set to $(|T| - i)$. For each tariff arc a_i , $i = \{1, \dots, |T|\}$, we define furthermore a fixed cost arc from the tariff arc a_i to the tariff arc a_{i+1} with cost 0. Finally, the cost of the arc (s, t) is set to $|T|$. See for example Figure 4.6, where $|T| = 5$.

For this type of network, if a client takes a tariff arc, the revenue from that tariff arc is at most 1. The cost of any path from s to t is bounded by the cost of the arc (s, t) . Clearly, the optimal solution for the leader induces the client to take as many tariff arcs as possible, i.e. the path through all tariff arcs, yielding a revenue of $|T|$.

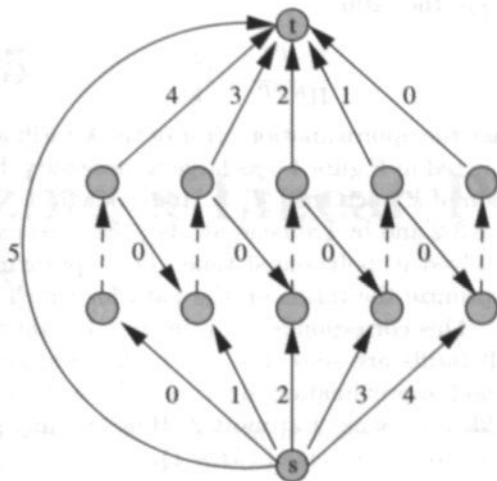


Figure 4.6: Worst-case network for k -tariff arcs algorithm ($|T| = 5, k \leq |T|$).

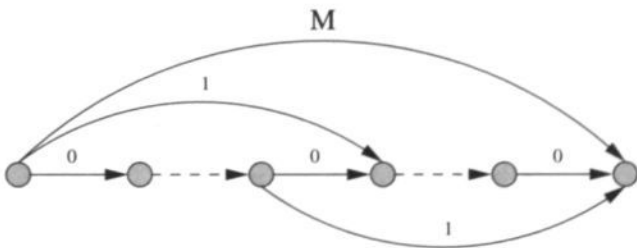


Figure 4.7: Worst-case network for null-path algorithm.

For the k -tariff arcs algorithm, the optimal solution will yield a revenue of k , since it is optimal to take as many tariff arcs as possible (at most k), each tariff arc yielding a revenue of 1.

If we denote the optimal revenue by Π^{OPT} and the revenue achieved by the k -tariff arcs algorithm by Π_{KTP} , the ratio

$$\frac{\Pi^{OPT}}{\Pi_{KTP}} = \frac{|T|}{k} \quad (4.12)$$

shows that we can make the approximation error of the k -tariff arcs algorithm on the type of network as depicted in Figure 4.6 as large as necessary, by taking $|T| = f(k)$, where $f(k)$ is a function of k , such that $|T|$ is large enough.

As shown in Section 3.2 and by Labbé et al. [41], the fixed path linear tarification problem can also be solved in polynomial time. One approximation algorithm we could consider is to optimize the tariffs on the path for which the upper bound on the revenue is highest. This corresponds to optimizing the tariffs on the path taken by the client when all tariffs are zero (the null-path). Figure 4.7 shows that this is indeed also not a good approximation algorithm. Indeed, optimizing the revenue over the null-path yields a revenue of at most 2. However, any path taking only one tariff arc can yield an optimal revenue of $(M - 1)$.

Chapter 5

Optimizing a Linear Pricing Strategy

In this chapter we describe single-level formulations of the network tariffication problem with multiple clients and linear pricing strategies. We consider a formulation using path variables, a formulation using arc variables, and a formulation using a combination of both arc and path variables. A branch and bound algorithm using path variables is furthermore proposed. Numerical results show the efficiency of applying the solution methods to the shortest path graph model instead of the original network.

5.1 Problem Formulations

Consider the following bilevel formulation of the network tariffication problem, as defined in Chapter 2. The upper level relates to the leader and fixes the tariffs, the lower level belongs to the clients, who given the tariffs set by the leader, determine their best (shortest) path.

$$\begin{aligned} \max \quad & \sum_{k \in K} \pi_{p_k^*}(d_k) \\ \text{s.t.} \quad & p_k^* \in \arg \min_{p \in P_k} \lambda_p(d_k) \quad \forall k \in K \end{aligned} \quad (5.1)$$

Since we assume demands are known a priori, both objective functions are linear in the tariffs and hence we are dealing with a linear-linear bilevel program. Notice furthermore that the bilevel program given by (5.1) is not polynomial in its input data, since the set of all possible paths for each client $k \in K$ may be exponential.

As we are considering only linear pricing strategies, recall from Chapter 2 that the cost of routing a demand on a tariff arc $a \in T$ is given by $\rho_a(d) = v_a d$. The cost of routing a demand on a fixed cost arc $a \in F$ is defined by $\delta_a(d) = c_a d$. Labbé et al.

[41] rewrite the path oriented formulation (5.1) to the following arc oriented bilevel model. Let b^k be the vector for each commodity $k \in K$ where for index i :

$$b_i^k = \begin{cases} -1 & \text{if } i = s_k, \\ +1 & \text{if } i = t_k, \\ 0 & \text{otherwise.} \end{cases} \quad (5.2)$$

and let the vector $y^k \in \mathbb{R}^{|A|}$ represent the flow on the arcs, for commodity $k \in K$. Define furthermore by i^+ , respectively i^- , the outgoing arcs, respectively, incoming arcs of node $i \in N$. Consider the following bilevel program for the network tarification problem, using arc-variables:

$$\begin{aligned} \max \quad & \sum_{k \in K} \sum_{a \in T} v_a d_k y_a^k \\ \min \quad & \sum_{k \in K} \left\{ \sum_{a \in T} v_a y_a^k + \sum_{a \in F} c_a y_a^k \right\} \\ \text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b_i^k \quad \forall k \in K, \forall i \in N \\ & y_a^k \geq 0 \quad \forall k \in K, \forall a \in A \\ & v_a \geq 0 \quad \forall a \in T \end{aligned} \quad (5.3)$$

The objective function of the upper level maximizes the revenue of the leader. The objective function of the lower level problem finds a shortest path in the network for each client. Both objective functions are bilinear, i.e. they are linear with respect to each of two variables. For the single-commodity case, primal-dual heuristics based on a penalization of the lower level objective function in (5.3) are proposed by Brotcorne et al. [13]. These heuristics were extended to multiple commodities in Brotcorne et al. [14].

5.1.1 Arc Formulation

Labbé et al. [41] have shown that it is possible to formulate the bilevel model described in (5.3) as a mixed integer program. To arrive at this formulation, note first that for fixed values of the tariffs v_a , $a \in T$, the lower level problem is a linear program. For each client $k \in K$, this linear program solves a shortest path problem. For the lower level problem, the optimal solution can be described by linear programming optimality conditions (primal and dual feasibility, and equality of the primal and dual objective function, see Chvátal [18]):

$$\begin{aligned}
\sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k &= b^k & \forall k \in K, \forall i \in N \\
\lambda_j^k - \lambda_i^k &\leq v_a & \forall k \in K, \forall a = (i, j) \in T \\
\lambda_j^k - \lambda_i^k &\leq c_a & \forall k \in K, \forall a = (i, j) \in F \\
\sum_{a \in T} v_a y_a^k + \sum_{a \in F} c_a y_a^k &= \lambda_{t_k}^k - \lambda_{s_k}^k & \forall k \in K \\
y_a^k &\geq 0 & \forall k \in K, \forall a \in A \\
v_a &\geq 0 & \forall a \in T
\end{aligned} \tag{5.4}$$

Replacing the lower level problem in (5.3) by its optimality conditions (5.4) yields the following (nonlinear) single-level formulation:

$$\begin{aligned}
\max \quad & \sum_{k \in K} \sum_{a \in T} v_a d_k y_a^k \\
\text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b^k & \forall k \in K, \forall i \in N \\
& \lambda_j^k - \lambda_i^k \leq v_a & \forall k \in K, \forall a = (i, j) \in T \\
& \lambda_j^k - \lambda_i^k \leq c_a & \forall k \in K, \forall a = (i, j) \in F \\
& \sum_{a \in T} v_a y_a^k + \sum_{a \in F} c_a y_a^k = \lambda_{t_k}^k - \lambda_{s_k}^k & \forall k \in K \\
& y_a^k \geq 0 & \forall k \in K, \forall a \in A \\
& v_a \geq 0 & \forall a \in T
\end{aligned} \tag{5.5}$$

The formulation given in (5.5) is bilinear, due to the product $v_a y_a^k$ in the objective function and the duality gap constraint. We can however linearize this term by introducing the variable v_a^k for all clients $k \in K$ and all tariff arcs $a \in T$. This variable is equal to $v_a y_a^k$ if and only if the client $k \in K$ takes the tariff arc $a \in T$. This relationship is achieved by the constraints:

$$\begin{aligned}
0 &\leq v_a^k \leq M y_a^k & \forall k \in K, a \in T \\
-M(1 - y_a^k) &\leq v_a^k - v_a \leq M(1 - y_a^k) & \forall k \in K, a \in T,
\end{aligned} \tag{5.6}$$

where the constant M is large enough. Note that the variables $y_a^k \in \{0, 1\}$ for all tariff arcs $a \in T$ and clients $k \in K$. This yields the following mixed integer program.

$$\begin{aligned}
\max \quad & \sum_{k \in K} \sum_{a \in T} v_a^k d_k \\
\text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b^k & \forall k \in K, \forall i \in N \\
& \lambda_j^k - \lambda_i^k \leq v_a & \forall k \in K, \forall a = (i, j) \in T \\
& \lambda_j^k - \lambda_i^k \leq c_a & \forall k \in K, \forall a = (i, j) \in F \\
& \sum_{a \in T} v_a^k + \sum_{a \in F} c_a y_a^k = \lambda_{t_k}^k - \lambda_{s_k}^k & \forall k \in K \\
& 0 \leq v_a^k \leq M y_a^k & \forall k \in K, a \in T \\
& -M(1 - y_a^k) \leq v_a^k - v_a \leq M(1 - y_a^k) & \forall k \in K, a \in T \\
& y_a^k \in \{0, 1\} & \forall k \in K, \forall a \in T \\
& y_a^k \geq 0 & \forall k \in K, \forall a \in F \\
& v_a \geq 0 & \forall a \in T
\end{aligned} \tag{5.7}$$

Recently, Dewez [25] considered valid inequalities for the formulation given in (5.7). Moreover, in Dewez [25] bounds for the M constant are computed for each constraint that are as small as possible. We will refer to the single-level mixed integer program of Labbé et al. [41], given in (5.7), in the remainder of this thesis as AMIP.

We can use the shortest path graph model of Section 2.5 in an arc oriented model by applying the formulation AMIP of Labbé et al. [41]. To this end, we define an SPGM for each client $k \in K$, as explained in Section 2.5. We create a multiple commodity mixed integer program by first allowing the tariffs on each tariff arc of the shortest path graph model to be different for each commodity, i.e. to allow for price discrimination and maximize the revenue of the leader over all commodities. We then prohibit price discrimination by setting tariffs on the arcs of each SPGM equal to each other if they represent the same tariff arc in the original network. We will refer to the adaptation of the formulation AMIP to the shortest path graph model as AMIP+.

5.1.2 Path Formulations

We rewrite the bilevel formulation given in (5.1) to the following single-level program, using path variables. To this end, we introduce the binary variable h_p which indicates whether or not a given path is taken in the optimal solution.

$$\begin{aligned}
\max \quad & \sum_{k \in K} \sum_{p \in P_k} \sum_{a \in T_p} h_p v_a d_k \\
\text{s.t.} \quad & \sum_{p \in P_k} h_p = 1 & \forall k \in K \\
& \lambda_p(d_k) \geq \sum_{q \in P_k} h_q \lambda_q(d_k) & \forall k \in K, \forall p \in P_k \\
& h_p \in \{0, 1\} & \forall k \in K, \forall p \in P_k \\
& v_a \geq 0 & \forall a \in T
\end{aligned} \tag{5.8}$$

The first constraint in formulation (5.8) indicates that a client selects only one path in the network, since the variables h_p are binary. The second constraint ensures that the path taken by the client is indeed the cheapest available path. Note that the bilevel formulation (5.8) is not only nonlinear in its objective function, but also in this constraint. We can linearize the formulation (5.8) by introducing the variable r_{pa} for each path $p \in P_k$, and tariff arc $a \in T$. The variable r_{pa} is equal to $v_a d_k$ if and only if client $k \in K$ takes the path $p \in P_k$ going through the tariff arc $a \in T$ and is equal to 0 otherwise. After adding the constraints enforcing this relationship, we can write the following mixed integer programming formulation. We will refer to this formulation as PMIP.

$$\begin{aligned}
\max \quad & \sum_{k \in K} \sum_{p \in P_k} \left(\sum_{a \in T_p} r_{pa} \right) \\
\text{s.t.} \quad & \sum_{p \in P_k} h_p = 1 & \forall k \in K \\
& \lambda_p(d_k) \geq \sum_{q \in P_k} \sum_{a \in F_q} h_q c_a d_k + \sum_{q \in P_k} \sum_{a \in T_q} r_{qa} & \forall k \in K, \forall p \in P_k \\
& \lambda_p(d_k) = \sum_{a \in F_p} c_a d_k + \sum_{a \in T_p} v_a d_k & \forall k \in K, \forall p \in P_k \\
& 0 \leq r_{pa} \leq h_p M & \forall k \in K, \forall p \in P_k, \forall a \in T \\
& -(1 - h_p)M \leq r_{pa} - v_a d_k \leq (1 - h_p)M & \forall k \in K, \forall p \in P_k, \forall a \in T_p \\
& h_p \in \{0, 1\} & \forall k \in K, \forall p \in P_k \\
& v_a \geq 0 & \forall a \in T
\end{aligned} \tag{5.9}$$

Note that for the PMIP, we are in theory still left with an exponential number of constraints. However, the remodeling of the network proposed in Section 2.5, combined with the model specific graph reduction methods, will allow us to generate the relevant paths for each commodity and thus reduce the necessary variables and constraints to a manageable amount.

Recently, Didi, Marcotte, and Savard [26] have proposed another single-level formulation using path variables. To arrive at this formulation, they introduce the variables Π^k and Λ^k for all clients $k \in K$. Π^k corresponds to the revenue per unit of demand to the leader from client $k \in K$ and Λ^k to the total cost per unit of demand of the path taken by client $k \in K$.

If for the client $k \in K$ we denote the path taken in the optimal solution by $p^* \in P_k$, we must have that:

$$\Pi^k = \sum_{a \in T_{p^*}} v_a \quad \text{and} \quad \Lambda^k = \Pi^k + \sum_{a \in F_{p^*}} c_a, \tag{5.10}$$

for the optimal solution. Since the clients can only take one path in the optimal solution we have that:

$$\Lambda^k = \Pi^k + \sum_{p \in P_k} h_p \sum_{a \in F_p} c_a \tag{5.11}$$

This yields the following single-level formulation, using path variables:

$$\begin{aligned}
 \max \quad & \sum_{k \in K} \sum_{p \in P_k} d_k \Pi^k \\
 \text{s.t.} \quad & \Pi^k \leq \sum_{a \in T_p} v_a + M_k(1 - h_p) \quad \forall p \in P_k, \forall k \in K \\
 & \sum_{a \in T_p} v_a + \sum_{a \in F_p} c_a - M_k(1 - h_p) \leq \Lambda^k \quad \forall p \in P_k, \forall k \in K \\
 & \Lambda^k \leq \sum_{a \in T_p} v_a + \sum_{a \in F_p} c_a \quad \forall p \in P_k, \forall k \in K \\
 & \Lambda^k = \Pi^k + \sum_{p \in P_k} h_p \sum_{a \in F_p} c_a \quad \forall k \in K \\
 & \sum_{p \in P_k} h_p = 1 \quad \forall k \in K \\
 & v_a \geq 0 \quad \forall a \in T \\
 & h_p \in \{0, 1\} \quad \forall k \in K, \forall p \in P_k
 \end{aligned} \tag{5.12}$$

For the formulation in (5.12), setting the values $h_p, p \in P_k$ to the optimal path for each client $k \in K$, yields the desired values of Π^k and Λ^k , given in (5.10).

5.1.3 Arc-Path Formulation

Didi et al. [26] have developed a single-level formulation for the network tariffication problem, that combines both arc and path variables. To arrive at their formulation, denote by I_a^k the set of paths from P_k that contain the arc $a \in A$:

$$I_a^k = \{p \in P_k | a \in p\}, \quad \forall a \in A, \forall k \in K. \tag{5.13}$$

Consider the single-level program AMIP using arc-variables given in (5.7). For AMIP the optimality conditions of the lower level program are the primal and dual feasibility and duality-gap constraints. Didi et al. [26] replace these constraints by the following optimality constraints:

$$\begin{aligned}
 \sum_{p \in P_k} h_p &= 1 & \forall k \in K \\
 \lambda_j^k - \lambda_i^k &= v_a - s_a & \forall k \in K, \forall a = (i, j) \in T \\
 \lambda_j^k - \lambda_i^k &= c_a - s_a & \forall k \in K, \forall a = (i, j) \in F \\
 0 \leq s_a &\leq M(1 - \sum_{p \in I_a^k} h_p) & \forall k \in K, \forall a \in A,
 \end{aligned} \tag{5.14}$$

where s_a is a slack variable for each arc $a \in A$ and the other variables are as defined previously for AMIP and PMIP. The first constraint in (5.14) ensures that the client only takes one path in the optimal solution. The other three constraints ensure that the path taken by the client is indeed the shortest path in the network. To see this, consider the last constraint in (5.14) and suppose the client $k \in K$ takes the shortest path p^* containing the arc $a^* \in A$. In that case, the corresponding slack variable s_{a^*} is equal to zero and the second and third constraint in (5.14) read, for all arcs on the path p^* :

$$\begin{aligned}\lambda_j^k - \lambda_i^k &= v_a \quad \forall k \in K, \forall a = (i, j) \in p^* \\ \lambda_j^k - \lambda_i^k &= c_a \quad \forall k \in K, \forall a = (i, j) \in p^*,\end{aligned}\quad (5.15)$$

For all arcs in the network the second and third constraints read:

$$\begin{aligned}\lambda_j^k - \lambda_i^k &\leq v_a \quad \forall k \in K, \forall a = (i, j) \in T \\ \lambda_j^k - \lambda_i^k &\leq c_a \quad \forall k \in K, \forall a = (i, j) \in F.\end{aligned}\quad (5.16)$$

The constraints in (5.15) and (5.16) show that the path p^* is indeed the shortest path in the network for the given costs and tariffs.

The flow on the arc $a \in A$ is equal to one if and only if there exists a path $p \in I_a^k$ such that $h_p = 1$. Furthermore, since a client can take only one path in the network, we have:

$$y_a^k = \sum_{p \in I_a^k} h_p \quad \forall k \in K, \forall a \in A. \quad (5.17)$$

Thus, for AMIP, the constraints (5.6) are furthermore replaced by the constraints:

$$\begin{aligned}0 \leq v_a^k &\leq M_a \sum_{p \in I_a^k} h_p \quad \forall k \in K, \forall a \in T \\ v_a^k &\leq v_a \quad \forall k \in K, \forall a \in T,\end{aligned}\quad (5.18)$$

where M_a , $a \in A$ are constants large enough. This yields the following single-level formulation, using arc and path variables.

$$\begin{aligned}\max \quad & \sum_{k \in K} \sum_{a \in T} v_a^k d_k \\ \text{s.t.} \quad & \sum_{p \in P_k} h_p = 1 \quad \forall k \in K \\ & \lambda_j^k - \lambda_i^k = v_a - s_a \quad \forall k \in K, \forall a = (i, j) \in T \\ & \lambda_j^k - \lambda_i^k = c_a - s_a \quad \forall k \in K, \forall a = (i, j) \in F \\ & 0 \leq s_a \leq M(1 - \sum_{p \in I_a^k} h_p) \quad \forall k \in K, \forall a \in A \\ & 0 \leq v_a^k \leq M_a \sum_{p \in I_a^k} h_p \quad \forall k \in K, \forall a \in T \\ & v_a^k \leq v_a \quad \forall k \in K, \forall a \in T \\ & h_p \in \{0, 1\} \quad \forall k \in K, \forall p \in P_k \\ & y_a^k \geq 0 \quad \forall k \in K, \forall a \in A \\ & v_a \geq 0 \quad \forall a \in T\end{aligned}\quad (5.19)$$

5.2 Branch and Bound

In this section we describe a branch and bound algorithm (see Ahuja, Magnanti, and Orlin [1]) for the network tariffication problem which uses the shortest path graph model from Section 2.5 and its reduction methods. This algorithm consists of two steps. In the first step we create for each client a shortest path graph model and apply to it the reduction methods of Section 2.5.2. For each client $k \in K$ we then find

the relevant shortest paths, denoted by P_k , by enumerating all paths in the shortest path graph model. In step two we solve the problem to optimality by a classical branch and bound method.

Recall from Chapter 2 the linear function $\lambda_p(d) = \gamma_p(d) + \pi_p(d)$ denoting the cost of a path p as a function of all tariff values. Let l_k be the path for client $k \in K$ with the smallest fixed cost, i.e., $l_k = \arg \min_{p \in P_k} \gamma_p(d_k)$ and u_k the path with the largest fixed cost, i.e., $u_k = \arg \max_{p \in P_k} \gamma_p(d_k)$. Note that u_k has no revenues for the leader, since it denotes the path with fixed cost arcs only. As defined earlier in Section 2.4, $\gamma_{u_k}(d) - \gamma_{l_k}(d)$ is an upper bound on the revenues that can be generated from client k . This is an important measure in the branch and bound algorithm.

5.2.1 Branching Rules

In each node of the branch and bound tree, we select a client, and we create a branch for each of the relevant paths of the client. The selection criterium of the clients is based on the upper bound on the revenue generated by each client for the leader: the client for which this upper bound is highest, is selected first. See Section 5.2.2 for the exact upper bound calculation used. Next, we walk through the search tree in a depth-first manner. See also Figure 5.1 for a graphical representation of the branching rules.

5.2.2 Node Processing

In each node of the branch and bound tree for some clients the path taken in the solution is fixed, whereas for other clients this choice is still to be made. In each node, we denote by the set $K_f \subseteq K$ the set of clients for which we have fixed the path taken in the solution. Suppose that for any client $k \in K_f$, the path taken in the optimal solution is the path p_k^* . We can find the optimal, revenue maximizing tariffs for the problem restricted to the clients in K_f by solving the following linear problem.

$$\begin{aligned} \max \quad & \sum_{k \in K_f} \pi_{p_k^*}(d_k) \\ \text{s.t.} \quad & \lambda_p(d_k) \geq \lambda_{p_k^*}(d_k) \quad \forall k \in K_f, \forall p \in P_k. \end{aligned} \tag{5.20}$$

The linear program described in (5.20) forces the path p_k^* to be the shortest path in P_k , while maximizing the leader's revenue.

We generate lower bounds in each node of the branch and bound tree by computing a feasible solution. Such a feasible solution can be created by solving (5.20) and then adding the revenues from the tariffs of (5.20) for the clients in $K \setminus K_f$. A better lower bound may be generated by fixing for all clients the path taken in the solution. For each client $k \in K_f$ we already know which path is taken in the solution and we denote it by p_k^* . For the clients $k \in K \setminus K_f$, we fix the path to the one for which the possible revenue is highest. This is the path with smallest fixed cost, i.e. the path l_k :

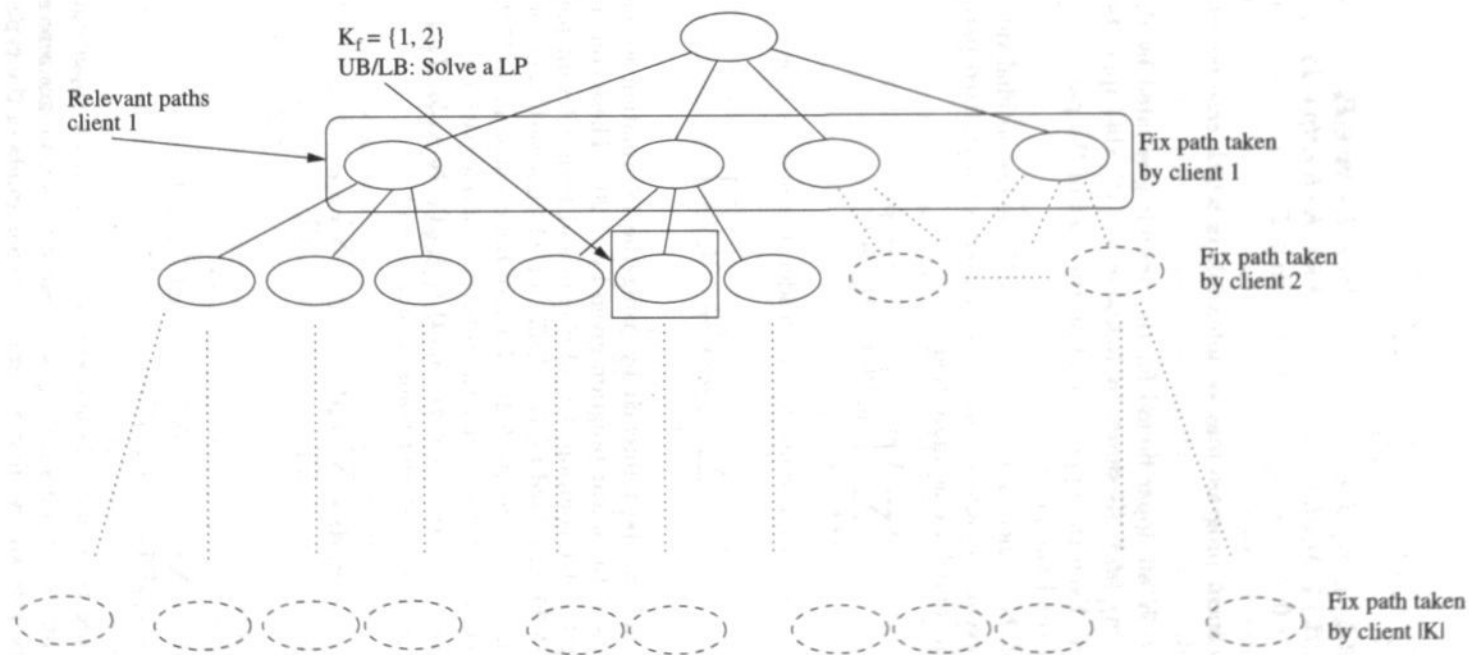


Figure 5.1: Branching rules for branch and bound algorithm.

$$\begin{aligned}
\max \quad & \sum_{k \in K_f} \pi_{p_k^*}(d_k) + \sum_{k \in K \setminus K_f} \pi_{l_k}(d_k) \\
s.t. \quad & \lambda_p(d_k) \geq \lambda_{p_k^*}(d_k) & \forall k \in K_f, \forall p \in P_k \\
& \lambda_p(d_k) \geq \lambda_{l_k}(d_k) & \forall k \in K \setminus K_f, \forall p \in P_k \\
& v_a \geq 0 & \forall a \in T
\end{aligned} \tag{5.21}$$

Note that this linear program may be infeasible in which case no sensible lower bound is generated.

For a client $k \in K$ an upper bound for the revenue generated by that client is given by $\gamma_{u_k}(d_k) - \gamma_{l_k}(d_k)$. As shown by Labbé et al. [41], this upper bound is not necessarily reached. Even the upper bound on the cost of the path, $\gamma_{u_k}(d)$, is not tight, as shown by the Example 2.1 of Chapter 2.

In each node of the branch and bound tree, let Π^* be the optimal value of (5.20), i.e., the optimal revenue obtained from the commodities with fixed paths. The remaining clients can contribute no more than

$$\sum_{k \in K \setminus K_f} \left\{ \gamma_{u_k}(d_k) - \gamma_{l_k}(d_k) \right\}. \tag{5.22}$$

An upper bound on the total revenue for the leader in a node is thus given by:

$$\Pi^* + \sum_{k \in K \setminus K_f} \left\{ \gamma_{u_k}(d_k) - \gamma_{l_k}(d_k) \right\} \tag{5.23}$$

This upper bound can be tightened by using the information we can retrieve from the constraints of the linear program given by (5.20). These constraints must be satisfied by any feasible solution, for all clients and their relevant paths. Thus, the constraints of (5.20) also hold for the clients for which a path is not fixed yet, as can be seen from the constraints in (5.21). Each of the constraints in (5.20) involves two paths, say p and q , and states that the total cost of one path (say p) is at most the total cost of the other path (say q): $\lambda_p(d) \leq \lambda_q(d)$. Now, let T_p be the set of tariff arcs on path p . The constraint reads in more detail

$$\gamma_p(d) + \sum_{a \in T_p} v_a d \leq \gamma_q(d) + \sum_{a \in T_q} v_a d \tag{5.24}$$

or equivalently,

$$\sum_{a \in T_p \setminus T_q} v_a - \sum_{a \in T_q \setminus T_p} v_a \leq \frac{\gamma_q(d) - \gamma_p(d)}{d} \tag{5.25}$$

If $\sum_{a \in T_q \setminus T_p} v_a$ can be bounded from above, then we have an upper bound on the sum of a set of tariffs. For instance, if $T_q \setminus T_p$ is empty, then we get zero as a trivial upper bound. Thus, this way we find a number of constraints of the type

$$\sum_{a \in T_i \subseteq T} v_a \leq b_i \quad (i \in I), \quad (5.26)$$

where b_i is a constant for $i \in I$. Each of the constraints in (5.26) gives an upper bound on the sum of the tariffs for the arcs in the set T_i , $i \in I$. Thus, if a client takes a path using a (sub)set of T_i , it is possible to upper bound the cost of taking that path for the client. Consider therefore a path $p \in P_k$ from commodity $k \in K \setminus K_f$, i.e., a commodity for which no path has been fixed yet. An upper bound on the total cost per demand was previously given by the path using only fixed arcs, i.e. the path γ_{u_k} . With the constraints of type (5.26) obtained from the clients in K_f and by defining $b_i = 0$ for $T_i = \emptyset$, the upper bound on the total cost of a path $p \in P_k$, denoted by $\Upsilon(p)$, is given by:

$$\Upsilon(p) = \min_{i \in I: T_p \subseteq T_i} (\gamma_p(d_k) + b_i d_k) \quad \forall k \in K \setminus K_f, \forall p \in P_k \quad (5.27)$$

For each commodity $k \in K$, the total cost of each path taken in a solution can be at most the minimum cost of all other possible paths for that commodity. Hence, an upper bound on the total cost of *any* path in P_k is given by:

$$\min_{p \in P_k} \Upsilon(p) \quad (5.28)$$

For each commodity $k \in K$, the upper bound on the revenue is still highest for the path with smallest fixed cost, i.e. the path l_k . We can thus improve the upper bound computation given in (5.23) by using the following upper bound:

$$\Pi^* + \sum_{k \in K \setminus K_f} \left\{ \min_{p \in P_k} \Upsilon(p) - \gamma_{l_k}(d_k) \right\} \quad (5.29)$$

We will use the upper bound calculation of (5.29) for our branch and bound algorithm.

5.3 Numerical Results

The branch and bound algorithm (PBB) was implemented in C++, using CPLEX 7.5 (see CPLEX [22]) to solve the linear programs as described in (5.20) in each node of the tree. The formulations AMIP, AMIP+ and PMIP were also implemented in C++, using CPLEX 7.5.

The solution methods developed in this chapter were tested on two types of data sets. The first type is a set of instances provided by France Télécom Research and Development and represents real life instances of the network tarification problem as occurring for France Télécom at the international level. In these instances, the graph represents a telecommunications network and the tariff arcs are the interconnections between the different operators. The clients are large corporations who wish to route their (international) demand on the network. A description of these real life instances is given in Table 5.1, where, for each data set, we describe the number of nodes and

arcs in the network, the number of tariff arcs and the number of clients. All tables can be found in Section 5.4.

The second type of data set consists of randomly generated graphs. A graph $G = (N, A)$ was generated by taking a random subset of the edges of the complete graph K_n . For each edge, we created two arcs, one for each direction between the two nodes. The tariff arcs are assigned randomly. The cost on each fixed cost arc was uniformly generated from $[20, 100]$. The source and destination of each demand were assigned randomly. The demand of each commodity was uniformly generated from $[1, 30]$. These instances are much larger than the real life instances, and are used to illustrate what happens with each algorithm when the number of (tariff) arcs or the number of clients in the network increases. Each of these data sets has a name of the form ' $xTyD$ ', where x represents the number of tariff arcs and y the number of clients in the network. Moreover, for the randomly created instances, each data set referred to as ' $xTyD$ ' actually consists of 5 instances. Statistics or execution times of a data set ' $xTyD$ ' are averages over all 5 instances created. Hence, each implemented formulation has been tested on a total of 165 random instances.

To illustrate the effectiveness of the shortest path graph model, Table 5.2 and Table 5.3 show some statistics on the number of paths generated for each data set using the shortest path graph model and its reduction methods for both type of data sets. In these two tables, the column MIN, resp. MAX, indicates the minimum, resp. maximum, number of paths generated over all commodities. The column AVRG gives the average number of paths generated for each commodity for the whole data set. As stated previously, in Table 5.3 the value of MIN, MAX and AVRG for each data set is an average over the 5 instances used. We can see from these two tables the competitiveness of the shortest path graph model and its reduction methods by the small number of paths generated on average for the given data sets. With this information we can expect the PBB algorithm and the PMIP formulation to benefit from this small number of paths for each commodity.

The performance of the shortest path graph model was tested using the PBB algorithm, the PMIP and AMIP+ formulations and compared to the mixed integer programming formulation AMIP of Labbé et al. [41], which uses the original arcs in the network and all tariff arcs available.

The computational results for the PBB algorithm, AMIP, AMIP+ and PMIP were established on an AMD Athlon 2400XP+ with 1 Gb of RAM, running Debian Linux 3.0 with kernel 2.4.18. For the AMIP+ formulation, the arcs generated by the shortest path graph model were used instead of the original arcs in the network. For the PMIP and the PBB algorithm, the size of the set P_k for each commodity $k \in K$ was determined by the number of paths generated by the shortest path graph model and its reduction techniques. As can be seen from Table 5.2 and Table 5.3, for all data sets this resulted on average in a small number of constraints for each commodity.

Table 5.4 gives an overview of the results for all algorithms for the real life instances. In this table, the column OPT indicates the value of the optimal solution for the given data set. The column CPU(s), resp. Nodes, indicates the execution time in seconds, resp. the number of nodes in the B&B tree for each algorithm. For the PBB algorithm, the PMIP and the AMIP+, the CPU time includes the time needed for the

generation of the shortest path graph model and all relevant paths. As can be seen in Table 5.4, the PBB algorithm efficiently uses the few relevant paths generated to find the optimal solution for each data set. When we compare its execution to the AMIP, we see that the execution time of the PBB algorithm is less or equal to the running time of CPLEX for the AMIP for each data set considered. The difference is especially large for the data sets D2, D3, D6, D7 and D8. The most striking difference is for the data set D3, for which the execution time of the AMIP is 113423 seconds, whereas the PBB algorithm takes only 30 seconds. The PMIP and AMIP+ columns show the efficiency and power of the shortest path graph model used. The execution time of the PMIP is 3 seconds or less for all data sets, except the data set D6. For this data set the execution time is however still much less than the execution time of the AMIP and less than the time needed for the PBB algorithm to execute. The AMIP+ column shows furthermore that the AMIP formulation can benefit very much from the shortest path graph model: for all data sets, the time needed for the AMIP+ formulation is a little more than the time needed for the PMIP formulation and much less than the time needed for the same formulation on the original network (AMIP).

Table 5.5 and Table 5.6 illustrate the behaviour of all formulations developed for randomly generated networks and show what happens when we increase the number of tariff arcs or clients in the network. For these instances, the CPU time of the PMIP, AMIP and AMIP+ does *not* include the time needed for the shortest path graph model. This value is given in the column SPGM. The maximal execution time for each solution method was set to 3600 seconds. If for a certain formulation or algorithm an instance could not be solved within this time, the column CPU indicates the number of instances where the time limit was exceeded, while the column Nodes gives the average percentual gap between the best (optimal) solution as found by any of the other algorithm within the time limit. All instances were solved to optimality by both the PMIP or AMIP+ formulation.

The numerical results for the random instances show that the performance of the algorithms developed is consistent with the numerical results for the real life instances: the PMIP and AMIP+ are the fastest, followed by respectively the PBB and AMIP. For the PMIP, AMIP+ and PBB algorithms, the numerical results show also that the time needed for the generation of the SPGM is the largest part of the execution time of the algorithm. When the size of the network increases, the shortest path graph model takes more time to generate, but its execution time is more dependent on the number of shortest paths that need to be calculated (depending on the number of tariff arcs or clients in the network) than on the size of the network each shortest path calculation is executed on. The execution time of the SPGM increases when the number of clients is increased, but is especially sensitive to the number of tariff arcs in the network. We can furthermore conclude that although the time needed for the SPGM generation increases when the number of tariff arcs or clients increases, the AMIP+ and PMIP formulation still use the information generated by the shortest path graph model very efficiently when compared to the PBB algorithm or AMIP formulation, where an increase in the number of clients in the network is soon a problem. For example, for the networks with 100 nodes and 2000 arcs, the

data set 90T50D, consisting thus of 90 tariff arcs and 50 clients, cannot be solved in 4 out of 5 instances for the PBB algorithm and 5 out of 5 instances for the AMIP formulation.

We can also note that although the PMIP and AMIP+ are much faster than the PBB algorithm and the AMIP, the advantage of the PBB algorithm over the PMIP, AMIP+ and AMIP is that it does not need a powerful mixed integer programming solver. Since very basic techniques have been used both with respect to concept and implementation, there is still room for improvement on this algorithm.

5.4 Tables

Table 5.1: Description of France Télécom instances.

Data	N	A	T	K
D1	29	92	6	13
D2	29	98	11	13
D3	43	176	10	20
D4	60	212	9	22
D5	60	212	10	8
D6	60	212	17	8
D7	60	212	21	18
D8	49	116	9	23
D9	33	116	15	30

Table 5.2: SPGM statistics for France Télécom instances.

Data	MIN	MAX	AVRG
D1	2	3	2.5
D2	2	5	3.1
D3	2	8	5.5
D4	2	15	4.2
D5	2	20	5.0
D6	2	30	12.0
D7	2	9	4.8
D8	2	10	4.0
D9	2	8	3.1

Table 5.3: SPGM statistics for random instances.

Networks with 100 nodes and 2000 arcs							
Data	MIN	MAX	AVRG	Data	MIN	MAX	AVRG
20T15D	2	2.8	2.1	90T20D	2	5.8	2.8
30T15D	2	3.4	2.2	90T30D	2	6.4	2.8
40T15D	2	3.6	2.3	90T40D	2	6.4	2.8
50T15D	2	4.0	2.6	90T50D	2	6.4	2.8
60T15D	2	5.0	2.8	90T60D	2	6.8	2.8
70T15D	2	5.4	3.1	90T70D	2	7.2	2.8
80T15D	2	5.4	3.1	90T80D	2	7.8	2.8
90T15D	2	7.2	3.4	90T90D	2	7.8	2.8
100T15D	2	7.2	3.5	90T100D	2	8.0	2.8
Networks with 75 nodes and 4000 arcs							
Data	MIN	MAX	AVRG	Data	MIN	MAX	AVRG
20T20D	2	3.6	2.2	100T30D	2	4.6	2.5
30T20D	2	4.0	2.3	100T40D	2	4.8	2.6
40T20D	2	4.2	2.4	100T50D	2	6.2	2.6
50T20D	2	4.4	2.5	100T60D	2	6.2	2.6
60T20D	2	5.0	2.6	100T70D	2	6.4	2.6
70T20D	2	5.4	2.8	100T80D	2	6.8	2.6
80T20D	2	5.8	2.9				
90T20D	2	6.0	3.0				
100T20D	2	6.8	3.1				

Table 5.4: CPU times for France Télécom instances.

		PMIP		PBB		AMIP+		AMIP	
Data	OPT	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>
D1	6.6225	< 1	1	< 1	52	1	8	2	187
D2	7.7928	< 1	10	1	201	< 1	49	8	1060
D3	728.435	1	55	30	14856	2	786	113423	13619345
D4	1321	1	78	37	21523	6	2445	208	10305
D5	995	1	6	< 1	185	1	16	1	35
D6	1426	35	134	39	1195	39	1246	399	71013
D7	1565	3	257	25	9689	8	1254	1426	76003
D8	664406	< 1	110	23	16709	1	32	317	29509
D9	189180	1	112	11	6129	1	132	56	3141

Table 5.5: CPU times for networks with 100 nodes and 2000 arcs.

	SPGM	PMIP		PBB		AMIP+		AMIP	
Data	<i>CPU(s)</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>
20T15D	3.4	0.2	2	0.2	164.8	0.4	5	38.6	121.2
30T15D	7	0.6	4.2	0.1	206.4	0.2	11.8	51.2	163.8
40T15D	11.6	0.2	2.2	0.4	215.6	0.4	3	142.2	934.2
50T15D	17.8	0.1	3.4	0.8	305.4	0.6	4.2	182.6	1105.8
60T15D	26.2	0.2	5	1.6	720.2	0.8	7.6	203.8	1096
70T15D	37	0.4	9.2	1.8	710.4	0.8	35.8	452.8	2868.4
80T15D	51.4	0.4	9.6	2.4	786	0.6	29.4	772.6	5763.6
90T15D	69	0.8	9	3.4	988.8	1.4	19.2	445.6	2876.4
100T15D	90	0.8	9	4.2	1038.4	1.2	29.2	754.2	5457.4
90T20D	90.4	0.6	2.8	2.6	555.2	1.6	9.4	856.6	3208.2
90T30D	134.2	0.8	12	69	12545.6	2.4	47.8	(3/5)	13.09%
90T40D	177.8	1.6	11	1243.6	206068.2	2.8	50.8	(4/5)	27.50%
90T50D	221.2	2.8	58.2	(4/5)	0.48%	3.6	137.4	(5/5)	26.14%
90T60D	264.6	3.6	54	(4/5)	3.77%	5.2	163	(5/5)	31.48%
90T70D	309	4.4	241.6	(5/5)	8.31%	7	386.4	(5/5)	28.47%
90T80D	352.8	5.2	378.4	(5/5)	10.85%	10	705.4	(5/5)	34.55%
90T90D	396.2	6.6	702.4	(5/5)	13.58%	22.6	2506.6	(5/5)	33.78%
90T100D	441	13	3168.6	(5/5)	16.39%	51.8	7460	(5/5)	35.44%

Table 5.6: CPU times for networks with 75 nodes and 4000 arcs.

	SPGM	PMIP		PBB		AMIP+		AMIP	
Data	<i>CPU(s)</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>	<i>CPU(s)</i>	<i>Nodes</i>
20T20D	9.4	0.4	8.2	0.8	477.4	0.4	6.2	466	960
30T20D	17.8	0.6	16.4	1.4	959.8	0.6	16.8	672.2	1245.2
40T20D	28.4	1	10.4	2	1128.8	1.2	14.8	1267.6	2909.2
50T20D	44.2	1	10	2.2	1143.2	1.4	15.4	1425.4	3330
60T20D	64.2	1.4	21.6	3.6	1385.4	1.4	20.4	(1/5)	4.41%
70T20D	90.6	1	23.4	4.8	1622.8	1.6	30.4	(3/5)	41.81%
80T20D	123.8	1.2	23	5.6	1759.8	1.8	32.6	(2/5)	21.30%
90T20D	163.8	1.6	20.6	6.4	1844.8	2.8	32.4	(3/5)	40.83%
100T20D	214.4	2.6	45.2	10.4	2499.2	3.6	25.8	(3/5)	39.40%
100T30D	316.6	3.2	8.6	46.8	9717	4.8	17.4	332.33	179.33
100T40D	420.4	4.6	27.4	706.4	114278.6	6.4	44	(2/5)	35.58%
100T50D	523.8	5.8	118.8	(3/5)	1.82%	7.4	166.6	(5/5)	22.18%
100T60D	627.8	7	1647.6	(5/5)	5.86%	8.6	192.8	(5/5)	26.26%
100T70D	732.4	12.8	3672.8	(5/5)	9.72%	12.2	1281.6	(5/5)	17.04%
100T80D	835.6	12.4	2142	(5/5)	15.21%	16.8	2369.6	(5/5)	17.38%

Chapter 6

Optimizing All Pricing Strategies

We extend the results obtained in Chapter 5, by considering pricing strategies which are commonly found in telecommunication networks. For the operator of a network, choosing a pricing strategy is an important part of his overall strategy. This decision may depend on (contradictory) economic and marketing factors: A pricing strategy that maximizes the revenue of the operator may not be very easy to understand for the user and thus not very easy to market.

Our model focuses on the economic aspect of the decision process. We show that it is possible to optimize the revenue of the operator for all pricing strategies considered. Using the results of numerical tests on a set of real life instances, we interpret the impact of each pricing strategy on the revenue for the operator and on the clients.

6.1 Model

We consider the variant of the network tarification problem, where the pricing strategy on the arcs is not restricted to a linear pricing strategy only (see Section 1.3.1 and Section 2.2). All pricing strategies are nonnegative, nondecreasing functions in the demand d denoted by the function $\rho_a(d)$ for all arcs $a \in T$. The function $\rho_a(d)$ on an arc $a \in T$ is a piecewise linear (not necessarily continuous) function. For all pricing strategies, the cost of routing no demand on the arc is zero.

The pricing strategies we consider are given in Figure 6.1. As we can see from this figure, for each pricing strategy, each linear piece of a pricing strategy must adhere to certain additional restrictions. For example, for the inclusive minutes pricing strategy (see Figure 6.1(c)), the pricing strategy has two linear pieces, where the first linear piece is constant in the demand d . For a formal definition of each pricing strategy, we refer to Section 2.2 on page 32.

The operator's optimization problem can be modelled by the following bilevel

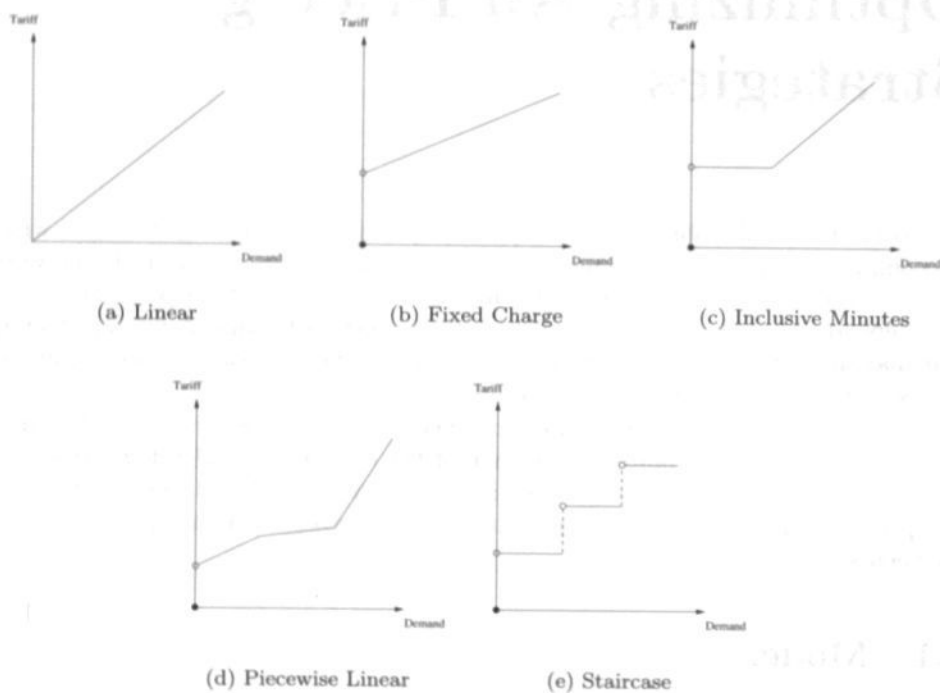


Figure 6.1: Pricing strategies.

formulation of the network tariffication problem using path variables.

$$\begin{aligned} \max \quad & \sum_{k \in K} \pi_{p_k^*}(d_k) \\ \text{s.t.} \quad & p_k^* = \arg \min_{p \in P_k} \lambda_p(d_k) \quad \forall k \in K. \end{aligned} \quad (6.1)$$

Since the demand of each client is known a priori, the revenue function $\pi_p(d)$ and the path length function $\lambda_p(d)$ in (6.1) are linear in the tariffs, for all pricing strategies. Formulation (6.1) is thus a linear-linear bilevel formulation.

The bilevel formulation (5.3) using arc variables, which holds for a linear pricing strategy only, can be adapted to hold for all pricing strategies as follows:

$$\begin{aligned} \max \quad & \sum_{k \in K} \sum_{a \in T} \rho_a(d_k) y_a^k \\ \min \quad & \sum_{k \in K} \left\{ \sum_{a \in T} \rho_a(d_k) y_a^k + \sum_{a \in F} \delta_a(d_k) y_a^k \right\} \\ \text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b_i^k \quad \forall k \in K, \forall i \in N \\ & y_a^k \geq 0 \quad \forall k \in K, \forall a \in A \end{aligned} \quad (6.2)$$

In this bilevel model, we use the same notation as in Chapter 5. The cost of routing a demand d on a tariff $a \in T$ is given by $\rho_a(d)$. The cost of routing a demand d on a fixed cost arc $a \in F$ is given by $\delta_a(d)$. The vector $y^k \in \mathbb{R}^{|A|}$ represents flow on the arcs, for commodity $k \in K$. For each node $i \in N$, b_i^k indicates whether the node is a supply or demand node.

We propose several methods to solve the network tariffication problem for all pricing strategies in Figure 6.1 to optimality. These methods are based on single-level formulations of (6.1) and (6.2) and on a generalization of the branch and bound algorithm proposed in Chapter 5.

6.2 Problem Formulations

The mixed integer problem (MIP) formulations of the network tariffication problem related to each pricing strategy that we propose are generalizations of the arc formulation AMIP and the path formulation PMIP described in Chapter 5.

6.2.1 Arc Formulation

The leader's and follower's objective function in (6.2) contain the (nonlinear) product $\rho_a(d_k) y_a^k$ for all arcs $a \in T$ and clients $k \in K$. Since the demands are known a priori, this product is bilinear: It is linear with respect to the flow variables and linear with respect to the tariffs. Whenever the pricing strategy is fixed, the lower level problem consists of solving $|K|$ independent shortest path problems. The bilevel formulation (6.2) can thus be written as a single level problem, by replacing the lower level linear program by its optimality conditions: primal and dual feasibility, and the duality

gap constraint (see Chvátal [18]). This yields the following single-level (nonlinear) formulation:

$$\begin{aligned}
\max \quad & \sum_{k \in K} \sum_{a \in T} \rho_a(d_k) y_a^k \\
\text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b_i^k & \forall k \in K, \forall i \in N \\
& \lambda_j^k - \lambda_i^k \leq \rho_a(d_k) & \forall k \in K, \forall a = (i, j) \in T \\
& \lambda_j^k - \lambda_i^k \leq \delta_a(d_k) & \forall k \in K, \forall a = (i, j) \in F \\
& \sum_{a \in T} \rho_a(d_k) y_a^k + \sum_{a \in F} \delta_a(d_k) y_a^k = \lambda_{t_k}^k - \lambda_{s_k}^k & \forall k \in K \\
& y_a^k \geq 0 & \forall k \in K, \forall a \in A
\end{aligned} \tag{6.3}$$

The formulation given in (6.3) is nonlinear, due to the product $\rho_a(d_k) y_a^k$ in the objective function and the duality gap constraint. Since the demand of each client is known a priori, we linearize this term by introducing the variable ρ_a^k for all tariff arcs $a \in T$ and all clients $k \in K$. The variable ρ_a^k is equal to $\rho_a(d_k)$ if and only if client $k \in K$ takes the tariff arc a . This relationship is achieved by the constraints:

$$\begin{aligned}
0 \leq \rho_a^k &\leq M y_a^k & \forall k \in K, a \in T \\
-M(1 - y_a^k) &\leq \rho_a^k - \rho_a(d_k) \leq M(1 - y_a^k) & \forall k \in K, a \in T,
\end{aligned} \tag{6.4}$$

where the constant M is large enough, and the variables $y_a^k \in \{0, 1\}$ for all tariff arcs $a \in T$ and clients $k \in K$. This yields the following mixed integer program.

$$\begin{aligned}
\max \quad & \sum_{k \in K} \sum_{a \in T} \rho_a^k \\
\text{s.t.} \quad & \sum_{a \in i^+} y_a^k - \sum_{a \in i^-} y_a^k = b_i^k & \forall k \in K, \forall i \in N \\
& \lambda_j^k - \lambda_i^k \leq \rho_a(d_k) & \forall k \in K, \forall a = (i, j) \in T \\
& \lambda_j^k - \lambda_i^k \leq \delta_a(d_k) & \forall k \in K, \forall a = (i, j) \in F \\
& \sum_{a \in T} \rho_a^k + \sum_{a \in F} \delta_a(d_k) y_a^k = \lambda_{t_k}^k - \lambda_{s_k}^k & \forall k \in K \\
& 0 \leq \rho_a^k \leq M y_a^k & \forall k \in K, a \in T \\
& -M(1 - y_a^k) \leq \rho_a^k - \rho_a(d_k) \leq M(1 - y_a^k) & \forall k \in K, a \in T \\
& y_a^k \in \{0, 1\} & \forall k \in K, \forall a \in T \\
& y_a^k \geq 0 & \forall k \in K, \forall a \in F
\end{aligned} \tag{6.5}$$

6.2.2 Path Formulation

The bilevel formulation (6.1) using path variables, can be written as the following single-level formulation:

$$\begin{aligned}
& \max \sum_{k \in K} \sum_{p \in P_k} h_p \pi_p(d_k) \\
& \text{s.t.} \quad \sum_{p \in P_k} h_p = 1 \quad \forall k \in K \\
& \quad \lambda_p(d_k) \geq \sum_{q \in P_k} h_q \lambda_q(d_k) \quad \forall k \in K, \forall p \in P_k \\
& \quad h_p \in \{0, 1\} \quad \forall k \in K, \forall p \in P_k
\end{aligned} \tag{6.6}$$

For the single-level formulation in (6.6), the binary variable h_p indicates whether or not a given path is taken in the optimal solution. The first constraint in (6.6) therefore guarantees that a client selects only one path. The second constraint ensures that the path taken by the client is indeed the cheapest available path. The demands are known a priori, thus the functions $\pi_p(d_k)$ and $\lambda_p(d_k)$ are linear in the tariffs. The products $h_p \pi_p(d_k)$ and $h_p \lambda_p(d_k)$ are therefore nonlinear. The formulation given in (6.6) is thus nonlinear in its objective function and in the second constraint, for all pricing strategies.

We linearize the formulation given in (6.6) by replacing the nonlinear product $h_p \rho_a(d_k)$ embedded in the objective function and in the second constraint by the variable r_{pa} for all paths $p \in P_k$ and tariff arcs $a \in T$ for a client $k \in K$. The variable r_{pa} is equal to $\rho_a(d_k)$ if and only if the client $k \in K$ takes the path $p \in P_k$ going through the tariff arc $a \in T$, and is equal to zero otherwise. This relationship is achieved by the following constraints:

$$\begin{aligned}
0 \leq r_{pa} \leq h_p M & \quad \forall k \in K, \forall p \in P_k, \forall a \in T \\
-(1 - h_p)M \leq r_{pa} - \rho_a(d_k) \leq (1 - h_p)M & \quad \forall k \in K, \forall p \in P_k, \forall a \in T_p,
\end{aligned} \tag{6.7}$$

where the constant M is large enough. We can thus write the bilevel formulation (6.1) as the following linear mixed integer program, for all pricing strategies.

$$\begin{aligned}
& \max \sum_{k \in K} \sum_{p \in P_k} \sum_{a \in T_p} r_{pa} \\
& \text{s.t.} \quad \sum_{p \in P_k} h_p = 1 \quad \forall k \in K \\
& \quad \lambda_p(d_k) \geq \sum_{q \in P_k} \sum_{a \in F_q} h_q \delta_a(d_k) + \sum_{q \in P_k} \sum_{a \in T_q} r_{qa} \quad \forall k \in K, \forall p \in P_k \\
& \quad \lambda_p(d_k) = \sum_{a \in F_p} \delta_a(d_k) + \sum_{a \in T_p} \rho_a(d_k) \quad \forall k \in K, \forall p \in P_k \\
& \quad 0 \leq r_{pa} \leq h_p M \quad \forall k \in K, \forall p \in P_k, \forall a \in T \\
& \quad -(1 - h_p)M \leq r_{pa} - \rho_a(d_k) \leq (1 - h_p)M \quad \forall k \in K, \forall p \in P_k, \forall a \in T_p \\
& \quad h_p \in \{0, 1\} \quad \forall k \in K, \forall p \in P_k
\end{aligned} \tag{6.8}$$

6.3 Branch and Bound

We show in this section that the branch and bound algorithm described in Chapter 5 can easily be adapted to hold for all pricing strategies studied. The algorithm consists of two steps: In the first step, the relevant paths are generated using the shortest path graph model of Section 2.5, while in the second step the network tariffication problem is solved to optimality by using a classical branch and bound algorithm.

6.3.1 Branching Rules

The branching rules apply already for all pricing strategies: In each node of the branch and bound tree, we select a client, and we create a branch for each of the relevant paths of the client. The selection method of the clients is based on the upper bound on the revenue generated by each client for the leader. The client for which this upper bound is highest, is selected first. Next, we walk through the search tree in a depth-first manner.

6.3.2 Node Processing

In each node of the branch and bound tree for some clients the path taken in the solution is fixed, whereas for other clients this choice is still to be made. Denote by the set $K_f \subseteq K$ the set of clients for which we have fixed the path taken in the solution. We generate lower bounds in each node of the branch and bound tree by computing a feasible solution. Such a feasible solution is created by fixing for all clients the path taken in the solution. For each client $k \in K_f$ we already know which path is taken in the solution and we denote it by p_k^* . For the clients $k \in K \setminus K_f$, we fix the path to the one for which the possible revenue is highest. This is the path with smallest fixed cost, i.e. the path l_k . The tariffs are determined using the following linear program:

$$\begin{aligned}
 \max \quad & \sum_{k \in K_f} \pi_{p_k^*}(d_k) + \sum_{k \in K \setminus K_f} \pi_{l_k}(d_k) \\
 \text{s.t.} \quad & \lambda_p(d_k) \geq \lambda_{p_k^*}(d_k) & \forall k \in K_f, \forall p \in P_k \\
 & \lambda_p(d_k) \geq \lambda_{l_k}(d_k) & \forall k \in K \setminus K_f, \forall p \in P_k
 \end{aligned} \tag{6.9}$$

Let Π^* be the optimal revenue obtained from the clients in the set K_f , whose path is already fixed. Π^* can be calculated for all pricing strategies using the following linear program:

$$\begin{aligned}
 \max \quad & \sum_{k \in K_f} \pi_{p_k^*}(d_k) \\
 \text{s.t.} \quad & \lambda_p(d_k) \geq \lambda_{p_k^*}(d_k) \quad \forall k \in K_f, \forall p \in P_k.
 \end{aligned} \tag{6.10}$$

The remaining clients can contribute no more than

$$\sum_{k \in K \setminus K_f} \left\{ \gamma_{u_k}(d_k) - \gamma_{l_k}(d_k) \right\}, \tag{6.11}$$

where $\gamma_{u_k}(d_k) - \gamma_{l_k}(d_k)$ is the upper bound on the revenue per client. An upper bound on the total revenue for the leader in a node is thus given by:

$$\Pi^* + \sum_{k \in K \setminus K_f} \left\{ \gamma_{u_k}(d_k) - \gamma_{l_k}(d_k) \right\} \quad (6.12)$$

In Chapter 5, the upper bound calculation in a node was refined specifically for a linear pricing strategy. For the other pricing strategies considered in this chapter, this upper bound refinement is invalid. It is furthermore difficult to find such a tailored upper bound calculation which holds for all pricing strategies. We will thus use the more general upper bound calculation (6.12).

6.4 Results

To obtain some insight in the consequences for the players in the network tarification problem (operator and clients) of using all the different pricing strategies defined, we try to answer the following questions. What is the best pricing strategy for the operator? And does a rise in revenue in case of a different pricing strategy mean that the clients pay more individually or does it mean that the number of clients generating revenue rises? To answer these questions, the MIP formulation (6.8) and the branch and bound algorithm proposed were implemented and tested on several real life instances obtained from France Télécom Research & Development. For the tests, the operator uses one of the pricing strategies studied and not a combination of all pricing strategies on the tariff arcs. Note furthermore that, for the pricing strategies studied, the number of linear pieces for each pricing strategy is at most 3.

For each pricing strategy, we denote each formulation (6.8) as follows. LMIP corresponds to a linear pricing strategy, FCMIP to a fixed charge pricing strategy, IMMIP to an inclusive minutes pricing strategy, SCMIP to a staircase pricing strategy and PLMIP to a piecewise linear pricing strategy on all the tariff arcs. Analogously, we denote for each pricing strategy each branch and bound algorithm as follows. LBB corresponds to a linear pricing strategy, FCBP to a fixed charge pricing strategy, IMBP to an inclusive minutes pricing strategy, SCBP to a staircase pricing strategy, and LSBB to a piecewise linear pricing strategy on all the tariff arcs.

In the remainder of this section, we first describe the data sets used, and give an overview of the running times of each MIP formulation and branch and bound algorithm for all data sets. Subsequently, we give some interpretations of the numerical results. All tables discussed can be found in Section 6.5.

6.4.1 Running Times

The MIP formulations using path variables and the branch and bound algorithms described in Section 6.2.2 and Section 6.3 were implemented in C++. The path MIP formulations and the branch and bound algorithm use the shortest path graph model in a preprocessing phase, to generate all the relevant paths. As shown in Section 2.5, the SPGM holds for all pricing strategies. As was noted in that section, additional

calculations are however necessary, since the inner graph of the SPGM is not identical for all clients in case of non linear pricing strategies. For the shortest path graph model, LEDA 4.1 (see LEDA [44]) was used. The path MIP formulations were solved using CPLEX 7.5 (see CPLEX [22]). The linear programs for the lower and upper bound calculations for the branch and bound algorithms were also solved using CPLEX 7.5. The formulations were tested on the data sets described in Table 6.1. These data sets represent telecommunications networks and are instances of interconnections markets, as described in Section 1.2.2. For each data set, we describe the number of nodes ($|N|$) and arcs ($|A|$) in the network, the number of tariff arcs ($|T|$) and the number of clients ($|K|$).

For Table 6.2, the column $(SPGM(s))$ represents the time in seconds necessary to generate the SPGM for all commodities, including the reduction methods, and to find all the relevant paths in the network. The last three columns in Table 6.2 represent the minimum (MIN), maximum (MAX) and average (AVRG) number of relevant paths generated by the SPGM. Table 6.2 shows that on average the number of relevant paths is very low, with data set D6 having the highest number of relevant paths on average, namely 7.4. Given this low number of paths for each data set, we can expect the MIP formulations to perform well for all pricing strategies. For data sets D5 and D7, the time to generate the SPGM and to find all relevant paths was the longest, namely 11.66 and 9.14 seconds. For all other data sets, the time needed to generate all relevant paths from the shortest path graph model was substantially less and in the order of 1 or 2 seconds.

The time required to solve each MIP formulation can be found in Table 6.3, for all pricing strategies. In this table, the column $CPU(s)$ indicates the number of seconds to solve the corresponding MIP formulation only. Hence, for the total execution time, we need to add the time spent on generating all the relevant paths from the shortest path graph model (see Table 6.2). The column *Nodes* gives the number of nodes in the branch and bound tree, as reported by CPLEX. For the branch and bound algorithms, the same information is available from Table 6.4.

Looking at the numerical results from Table 6.3 and Table 6.4, we see that the MIP formulations outperform the branch and bound algorithms by far for all data sets except D1, D2 and D3. In general, the MIP formulations perform very well on all data sets and for all pricing strategies. The MIP formulations enable us to solve the network tariffication problem for almost all data sets and pricing strategies in a few seconds. The most surprising result is that the data set D5 appears to be the most difficult one to solve. Apparently data set D5 has some other intrinsic difficulty than the number of paths in the network, since the average number of paths found is relatively low (lower than for either D4, D6, D7 or D8).

6.4.2 The Best Pricing Strategy

To identify the best pricing strategy for the operator of a network, consider Table 6.5, where the exact revenue to the operator for all pricing strategies and all data sets is stated. For convenience, the highest revenue is given in bold letters for each data set. As we can see from Table 6.5, the revenue to the operator always rises in case of a

fixed charge or piecewise linear pricing strategy. This need not be the case however for the inclusive minutes or staircase pricing strategy. Table 6.5 illustrates one of the shortcomings of our MIP formulations and branch and bound algorithms, in the sense that we do not optimize the size of each linear piece: As such, the inclusive minutes pricing strategy is not a true generalization of the linear and fixed charge pricing strategy, and can generate less revenue. Especially the staircase pricing strategy is very restrictive due to the number of linear pieces and the constant tariff in each linear piece. A staircase pricing strategy yields optimal revenues which are always lower than all other pricing strategies, for all data sets considered. Table 6.5 shows that the piecewise linear pricing strategy always yields the highest revenue. The linear by segments pricing strategy contains all other pricing strategies as a special case, even when the size of each linear piece is not optimized. It thus should indeed yield at least as high a revenue as the other pricing strategies. Besides the best pricing strategy, the numerical results and previous observations allow us to state the worst pricing strategy: A staircase pricing strategy will generate less revenue to the operator. This property could be changed by optimizing the number and the size of the linear pieces.

6.4.3 Consequences for the Clients

Table 6.6 is useful to investigate the consequences of the choice of pricing strategy for the clients. It shows the number of clients taking at least one tariff arc (and thus generating revenue to the operator) for all data sets and pricing strategies. That same number is given as a percentage of all clients in the network between parentheses. Table 6.6 thus also gives the percentage of all clients served for all pricing strategies and all data sets. Table 6.7 combines the information from Table 6.5 and Table 6.6 and shows the average revenue to the operator, indexed on a linear pricing strategy. From these tables, we can see that the number of clients served stays fairly constant for each data set, with the exception once again for the staircase pricing strategy. The percentage of clients served varies across the different data sets from 20% of all clients to 100%. From Table 6.5 and Table 6.7, we see that although the revenue to the operator rises with at most 10%, when taking a fixed charge or inclusive minutes pricing strategy instead of a linear pricing strategy, the average revenue per client rises with at most 20% in that case. As we have seen previously from Table 6.5, the revenue decreases significantly for a staircase pricing strategy. The average revenue per client however, stays fairly constant for that particular pricing strategy, with the exception of data set D3. For this data set, the average revenue per client more than doubles when compared to the linear pricing strategy. Apparently, for this data set, approximately half of the clients yield little revenue and are dropped for the bigger clients when taking a staircase pricing strategy. Note that although the average revenue per client is higher, the total revenue is however still approximately 10% lower than for a linear pricing strategy. If we disregard the staircase pricing strategy, then we can say that clients pay more on average for a different pricing strategy than the linear pricing strategy. This especially holds for the fixed charge and inclusive minutes pricing strategy.

6.4.4 Conclusion

As we have seen in the previous section, choosing a pricing strategy is as important as optimizing the pricing strategy: Depending on the particular market to serve, one pricing strategy may yield more revenue than the others. From the analysis of the numerical results we can conclude that the piecewise linear pricing strategy yields the highest revenue. We can furthermore conclude from Section 6.4.2 that the staircase pricing strategy is very restrictive and yields, on average, less revenue to the leader than the other pricing strategies for the test instances considered. For most data sets studied, the staircase pricing strategy furthermore perturbed the solution much more when compared to the other pricing strategies: the average revenue per client and the percentage of clients served may differ very much.

Although we studied the best pricing strategy from a revenue point of view, the choice of a pricing strategy by a telecommunications operator may not only depend on the revenue, but also on the transparency for the user: Although a piecewise linear pricing strategy yields more revenue to the operator, an inclusive minutes pricing strategy is easier to understand. As such, further research could study the optimization of the size of each linear piece for a particular pricing strategy. Another interesting aspect would be to study the optimization of the number of such linear pieces. Especially for a staircase pricing strategy, this could be beneficial to the operator. Such research would however need a different optimization model than the one given here.

6.5 Tables

Table 6.1: Description of France Télécom instances.

Data	$ N $	$ A $	$ T $	$ K $
D1	29	92	6	10
D2	29	98	11	15
D3	43	176	10	20
D4	49	116	9	23
D5	33	116	15	30
D6	63	174	13	22
D7	142	388	29	28
D8	51	148	13	28

Table 6.2: SPGM generation and path number statistics.

Data	SPGM(s)	MIN	MAX	AVRG
D1	0.07	2	5	2.9
D2	0.27	2	3	2.2
D3	0.42	2	10	3.1
D4	0.59	2	10	4.3
D5	11.66	2	8	3.1
D6	0.95	3	12	7.4
D7	9.14	2	15	5
D8	1.87	2	24	4.6

Table 6.3: CPU times for path MIP formulations.

	LMIP		FCMIP		IMMIP		SCMIP		PLMIP	
Data	CPU(s)	Nodes	CPU(s)	Nodes	CPU(s)	Nodes	CPU(s)	Nodes	CPU(s)	Nodes
D1	0.01	0	0.01	0	0.01	0	0.06	20	0.05	0
D2	0.01	0	0.01	0	0.01	0	0.07	2	0.03	0
D3	0.06	32	0.14	71	0.2	88	0.03	48	0.16	88
D4	0.33	93	0.94	364	0.55	152	0.76	391	0.47	155
D5	82.91	24889	2103.25	523448	87.52	22043	1152.72	382707	419.44	116026
D6	1.15	193	2.13	215	2.24	371	5.63	1675	2.24	239
D7	0.77	188	0.61	160	0.34	107	7.88	5237	0.24	65
D8	0.31	77	0.57	116	1.07	345	3.11	1417	0.31	16

Table 6.4: CPU times for branch and bound algorithms.

	LBB		FCBB		IMBB		SCBB		LSBB	
Data	CPU(s)	Nodes	CPU(s)	Nodes	CPU(s)	Nodes	CPU(s)	Nodes	CPU(s)	Nodes
D1	0.01	11	0.03	24	0.01	1	0.35	468	0.01	1
D2	0.02	28	0.04	30	0.1	62	0.12	126	0.06	28
D3	1.48	1185	5.33	2377	6.29	2377	6.42	4042	7.57	2377
D4	42.14	24539	654.61	270140	349.21	122727	330.54	140088	367.27	93956
D5	888.21	173644	28585.02	4129908	32648.09	3711343	9851.38	1261459	18139.05	1698082
D6	885.15	307225	6322.36	1427860	12658.2	2046394	2961.41	637111	10557.57	1526066
D7	1607.91	384854	3137.45	574414	1206.03	174994	3796.06	665281	496.08	61624
D8	123.21	53879	556.79	159217	1014.3	254482	695.05	198180	694.45	153129

Table 6.5: Revenue per pricing strategy.

	LMIP	FCMIP	IMMIP	SCMIP	PLMIP
D1	71.6092	71.6092	74.4482	43.3891	74.4482
D2	20.7897	20.7897	20.5143	17.9385	20.7897
D3	500.997	511.304	511.304	452.952	511.304
D4	775918	789149	837676	743835	847623
D5	1051380	1052420	1044330	956888	1068800
D6	819.750	827.512	819.756	702.730	828.261
D7	90348	91970	93516	73049	95101
D8	437.250	465.725	459.702	392.190	475.692

Table 6.6: Number and percentage of clients served.

	LMIP		FCMIP		IMMIP		SCMIP		PLMIP	
D1	10	(100%)	10	(100%)	10	(100%)	7	(70%)	10	(100%)
D2	13	(87%)	13	(87%)	13	(87%)	12	(80%)	13	(87%)
D3	9	(45%)	9	(45%)	9	(45%)	4	(20%)	9	(45%)
D4	18	(78%)	16	(70%)	16	(70%)	17	(74%)	20	(87%)
D5	29	(97%)	29	(97%)	25	(83%)	25	(83%)	28	(93%)
D6	14	(64%)	13	(59%)	13	(59%)	16	(73%)	13	(59%)
D7	21	(75%)	21	(75%)	22	(79%)	21	(75%)	25	(89%)
D8	20	(71%)	20	(71%)	20	(71%)	20	(71%)	20	(71%)

Table 6.7: Average revenue per client (indexed).

	LMIP	FCMIP	IMMIP	SCMIP	PLMIP
D1	100	100	104	87	104
D2	100	100	99	93	100
D3	100	102	102	203	102
D4	100	114	121	102	98
D5	100	100	115	106	105
D6	100	109	108	75	109
D7	100	102	99	81	88
D8	100	107	105	90	109

Chapter 7

Summary and Concluding Remarks

In Chapter 1 and Chapter 2 we introduce the network tarification problem (NTP). A remodeling of the network is proposed which tries to focus on the combinatorial structure of the problem: the shortest path graph model (SPGM). Besides providing additional insight into the network tarification problem, the shortest path graph model is used in later chapters to solve the network tarification problem to optimality. The shortest path graph model prompted a patent application by France Télécom, which was granted by the *Institut National de la Propriété Industrielle* in Paris, France on August 15th, 2003 (see [9]) and by the *World Intellectual Property Organization* on August 21st, 2003 (see [8]).

The complexity of the network tarification problem is studied in Chapter 3. The network tarification problem is shown to be \mathcal{NP} -hard, using a proof by Roch et al. [53]. We show that two special cases of the network tarification problem, the river tarification problem and the all-service network tarification problem, remain \mathcal{NP} -hard. We identify several new polynomially solvable special cases. One of the new special cases introduced is the bounded arcs linear tarification problem. For this special case, we are dealing with a linear pricing strategy on the tariff arcs, where the number of tariff arcs is bounded a priori. Another important special case is the parametric tarification problem, where the pricing strategy on the tariff arcs is dependent on a single parameterized tariff only. Parametric tarification itself has an interesting special case, uniform tarification. As shown in Chapter 4, a chapter dedicated to approximation results, uniform tarification allows for a $|T|$ -approximation algorithm for the river tarification problem with a linear pricing strategy on the tariff arcs, where $|T|$ is the number of tariff arcs in the network. Whenever the costs on the fixed cost arcs in the network are polynomially bounded in terms of $|T|$, we can improve this bound and show that uniform tarification provides an $\mathcal{O}(\log(|T|))$ -approximation. In Chapter 4, we show that the all-service network tarification problem is inapproximable with $\mathcal{O}(|T|^{1-\varepsilon})$ and with $\mathcal{O}(|K|^{1/2-\varepsilon})$, unless $\mathcal{ZPP} = \mathcal{NP}$.

An interesting avenue for further research would be to extend the class of polynomially solvable cases even further. One special case which could be interesting and extends the research presented is to bound the number of commodities a priori. Furthermore, the complexity of the network tarification problem with a non linear pricing strategy and multiple tariff arcs, where the number of tariff arcs is upper bounded a priori is unknown. It remains open at this point if there exists a better approximation result for the river tarification problem or if we can extend our result to the network tarification problem with multiple clients. It is furthermore unknown at this point if the network tarification problem is inapproximable without the all-service assumption.

In Chapter 5 we consider exact methods for the optimization of the network tarification problem for a linear pricing strategy on the tariff arcs. To this end, several mixed integer programming (MIP) formulations and a branch and bound algorithm are proposed. Although, as shown in Chapter 2, the upper bound on the number of undominated paths can be reached for artificial instances, the numerical results show the strength of the shortest path graph model for the instances tested: On average only a small number of relevant (undominated) paths are generated by the shortest path graph model. The strength of the shortest path graph model is furthermore shown by an existing arc formulation of the problem that benefits from the remodeling of the network. The MIP formulations and the branch and bound algorithm proposed in Chapter 5 are generalized for all pricing strategies studied in Chapter 6. The impact of each pricing strategy on the revenue for the leader and the consequences for the clients is investigated, using the numerical results on a set of real life instances.

It is shown in Chapter 5 and Chapter 6 that the shortest path graph model is very helpful in remodeling the network and that formulations based on this remodeled network are very efficient. As a possible future extension, the paths generated by the shortest path graph model could be used in other models based on a path formulation of the network tarification problem. Recently, Didi et al. [26] have proposed a new MIP formulation using path variables and a new MIP formulation using a combination of arc and path variables for the network tarification problem with linear pricing strategies. These new formulations have not been implemented and tested, but it would be interesting to see how they compare to the path formulation proposed. Another extension would be to consider the network tarification problem with capacities on the arcs. Finally, the optimization of the size and number of linear pieces in a pricing strategy could be an interesting avenue for further research.

Bibliography

- [1] R. Ahuja, T. Magnanti, and J. Orlin. *Network flows*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1993.
- [2] B. Aspvall and R.E. Stone. Khachiyan's linear programming algorithm. *Journal of Algorithms*, 1(1):1–13, 1980.
- [3] J. Bard. An investigation of the linear three level programming problem. *IEEE Transactions on Systems, Man, and Cybernetics*, 14:711–717, 1984.
- [4] J. Bard and J. Falk. An explicit solution to the multi-level programming problem. *Computers and Operations Research*, 9:77–100, 1982.
- [5] W. Ben-Ameur, N. Michel, B. Liau, J. Geffard, and E. Gourdin. Routing strategies for IP networks. *Teletronikk*, 2/3:145–158, 2001.
- [6] W. Bialas and M. Karwan. On two-level optimization. *IEEE Transactions on Automatic Control*, 27:211–214, 1982.
- [7] M. Bouhtou, M. Diallo, and L. Wynter. *Fair Network Resource Allocation and Link Pricing: A Numerical Study*. Optimization and Optimal Control. P.M. Pardalos and I. Tsevendorj and R. Enkhbat eds., World Scientific Publishing, 2003.
- [8] M. Bouhtou, S. van Hoesel, and A. F. van der Kraaij. Method and system for adjusting the control parameter values of a network in an optimal manner. *World Intellectual Property Organization*, International Patent(WO 03/069846 A1), August 21st 2003.
- [9] M. Bouhtou, S. van Hoesel, and A. F. van der Kraaij. Procédé et système pour ajuster des valeurs de paramètres de contrôle d'un réseau de façon optimale. *Institut National de la Propriété Industrielle*, France Télécom Patent(FR 2 835 988 A1), August 15th 2003.
- [10] M. Bouhtou, S. van Hoesel, A. F. van der Kraaij, and J.L. Lutton. Tariff optimization in networks. *Meteor Research Memorandum*, RM03011, 2003.
- [11] J. Bracken and J. McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21:37–44, 1973.

- [12] D. Braess. Über ein paradoxon der verkehrsplanung. *Unternehmensforschung*, 1968.
- [13] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard. A bilevel model and solution algorithm for a freight tariff-setting problem. *Transportation Science*, 34(3):289–302, 2000.
- [14] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard. A bilevel model for toll optimization on a multicommodity transportation network. *Transportation Science*, 35(4):345–358, 2001.
- [15] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard. Joint design and pricing on a network. *Submitted for publication*, 2003.
- [16] R.E. Burkard, V.G. Deineko, R. van Dal, J.A.A. van der Veen, and G.J. Woeginger. Well-solvable special cases of the tsp: A survey. SFB-Report 52, Technische Universität Graz, December 1995.
- [17] W. Candler and R. Townsley. A linear two-level programming problem. *Computers and Operations Research*, 9:59–76, 1982.
- [18] V. Chvátal. *Linear Programming*. W.H. Freeman and Company, New York, 1983.
- [19] R. Cole, Y. Dodis, and T. Roughgarden. How much can taxes help selfish routing? In *Proceedings of the 4th Annual ACM Conference on Electronic commerce*, pages 98–107, 2003.
- [20] R. Cole, Y. Dodis, and T. Roughgarden. Pricing network edges for heterogeneous selfish users. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 521–530, 2003.
- [21] J.P. Côté, P. Marcotte, and G. Savard. A bilevel modeling approach to pricing and fare optimization in the airline industry. *Journal of Revenue and Pricing Management*, 2:23–36, 2003.
- [22] CPLEX. ILOG CPLEX. v7.5, 2004.
- [23] G. Dantzig, R. Fulkerson, and S. Johnson. Solution of a large-scale travelling-salesman problem. *Operations Research*, 2:393–410, 1954.
- [24] S. Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52:333–359, 2003.
- [25] S. Dewez. *On the Toll Setting Problem*. PhD thesis, Université Libre de Bruxelles, Institut de Statistique et de Recherche Opérationnelle, 2004.
- [26] M. Didi, P. Marcotte, and G. Savard. On path formulations of the toll setting problem. Working paper, Université de Montréal, 2004.
- [27] EU Directive. Directive 98/10/EC. European Union, 1998.

- [28] J. Fortuny-Amat and B. McCarl. A representation and economic interpretation of a two-level programming problem. *Journal of the Operational Research Society*, 32:783–792, 1981.
- [29] B. Fortz and M. Thorup. Increasing internet capacity using local search. Technical Report IS-MG 2000/21, Université Libre de Bruxelles, 2000. To appear in *Computational Optimization and Applications*.
- [30] B. Fortz and M. Thorup. Internet traffic engineering by optimizing OSPF weights. In *Proc. 19th IEEE Conf. on Computer Communications (INFOCOM)*, pages 519–528, 2000.
- [31] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, San Francisco, 1979.
- [32] A. Grigoriev, S. van Hoesel, A. F. van der Kraaij, M. Uetz, and M. Bouhtou. Pricing network edges to cross a river. *Meteor Research Memorandum*, RM04009, 2004.
- [33] P. Hansen, B. Jaumard, and G. Savard. New branch and bound rules for linear bilevel programming. *SIAM Journal on Scientific and Statistical Computing*, 13(5):1194–1217, 1992.
- [34] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica*, 182:105–142, 1999.
- [35] O. Jahn, R. Möhring, and A. Schulz. Optimal routing of traffic flows with length restrictions in networks with congestion. *Operations Research Proceedings*, pages 437–442, 1999.
- [36] R.G. Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Mathematical Programming*, 32:146–164, 1985.
- [37] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.
- [38] R.M. Karp and J. B. Orlin. Parametric shortest path algorithms with an application to cyclic staffing. *Discrete Applied Mathematics*, 3:37–45, 1981.
- [39] L.G. Khachian. A polynomial algorithm for linear programming. *Doklady Akad. Nauk USSR*, 244(5):1093–96, 1979. Translated in *Soviet Math. Doklady*, 20, 191–194.
- [40] C. Kolstad and L. Lasdon. Derivative evaluation and computational experience with large bilevel mathematical programs. *Journal of Optimization Theory and Applications*, 65:485–499, 1990.
- [41] M. Labbé, P. Marcotte, and G. Savard. A bilevel model of taxation and its application to optimal highway pricing. *Management Science*, 44:1608–1622, 1998.

- [42] M. Labbé, P. Marcotte, and G. Savard. *On a class of bilevel programs*, pages 183–206. *Nonlinear Optimization and Related Topics*. Di Pillo and Giannessi eds., Kluwer Academic Publishers, 1999.
- [43] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy-Kan, and D.B. Shmoys. *The Travelling Salesman Problem*. John Wiley & Sons, New York, 1985.
- [44] LEDA. Algorithmic Solutions Software GmbH, Library for Efficient Data Types and Algorithms. LEDA 4.1, 2004.
- [45] L.N. Liu and J.F. McDonald. Economic efficiency of second-best congestion pricing schemes. *Transportation Research-B*, 33(2):157–188, 1999.
- [46] Z.Q. Luo, J.S. Pang, and D. Ralph. *Mathematical Programs With Equilibrium Constraints*. Cambridge University Press, The Edinburgh Building, Cambridge CB2 2RU, UK, 1996.
- [47] J. MacKie-Mason and H. Varian. Pricing congestible network resources. *IEEE Journal on Selected Areas in Communications*, 13(7):1141–1149, 1995.
- [48] P. Marcotte and G. Savard. *Bilevel programming: Formulations, applications, algorithms*, volume 1 of *Encyclopedia of Optimization*. C.A. Floudas and P.M. Pardalos eds., Kluwer Academic Publishers, Dordrecht, 2001.
- [49] P. Marcotte and G. Savard. *A bilevel programming approach to price setting*, pages 97–117. *Decision and Control in Management Science. Essays in Honor of Alain Haurie*. G. Zaccour ed., Kluwer Academic Publishers, 2002.
- [50] P. Marcotte, G. Savard, and F. Semet. A bilevel programming approach to the travelling salesman problem. *Operations Research Letters*, 32:240–248, 2004.
- [51] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, The Edinburgh Building, Cambridge CB2 2RU, UK, 1995.
- [52] C.H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization, Algorithms and Complexity*. Dover Publications, Inc., 31 East 2nd Street, Mineola, N.Y. 11501, 1998.
- [53] S. Roch, G. Savard, and P. Marcotte. Design and Analysis of an approximation algorithm for Stackelberg network pricing. *Optimization Online*, 2003.
- [54] T. Roughgarden. Designing networks for selfish users is hard. In *Proceedings of the 42nd Annual IEEE Symposium on the Foundations of Computer Science*, pages 472–482, 2001.
- [55] T. Roughgarden. How unfair is optimal routing? In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 203–204, 2002.
- [56] T. Roughgarden and É. Tardos. How bad is selfish routing? *Journal of the Association for Computing Machines*, 49(2):236–259, 2002.

- [57] G. Savard and J. Gauvin. The steepest descent direction for the nonlinear bilevel programming problem. *Operations Research Letters*, 15:275–282, 1994.
- [58] K. Shimizu, Y. Ishizuka, and J.F. Bard. *Nondifferentiable and Two-Level Mathematical Programming*. Kluwer Academic Publishers, Boston, 1997.
- [59] H. Stackelberg. *The Theory of the Market Economy*. Oxford University Press, New York, 1952.
- [60] W.R. Stevens. *TCP/IP Illustrated, Volume 1: The Protocols*. Addison Wesley, 1st edition, January 1994.
- [61] J. Tirole. *The Theory of Industrial Organization*. MIT Press, Cambridge, Massachusetts, 1997.
- [62] US Act. Telecommunications reform act. United States of America, 1996.
- [63] S. van Hoesel, A. F. van der Kraaij, C. Mannino, G. Oriolo, and M. Bouhtou. Polynomial cases of the tariffication problem. *Meteor Research Memorandum*, RM03053, 2003.
- [64] H.R. Varian. *Microeconomic Analysis*. W.W. Norton & Company, Inc., 500, Fifth Avenue, New York, N.Y. 10110, 3rd edition, 1992.
- [65] E.T. Verhoef. Second-best congestion pricing in general networks. Heuristic algorithms for finding second-best optimal toll levels and toll points. *Transportation Research-B*, 36(8):707–729, 2002.
- [66] L. Vicente, G. Savard, and J. Júdice. Descent approaches for quadratic bilevel programming. *Journal of Optimization Theory and Applications*, 81:379–399, 1994.
- [67] L.N. Vicente. *Bilevel programming: Introduction, history and overview*, volume 1 of *Encyclopedia of Optimization*, pages 178–180. C.A. Floudas and P.M. Pardalos eds., Kluwer Academic Publishers, Dordrecht, 2001.
- [68] L.N. Vicente and P. H. Calamai. Bilevel and multilevel programming: A bibliography review. *Journal of Global Optimization*, 5:291–306, 1994.
- [69] N. E. Young, R. E. Tarjan, and J. B. Orlin. Faster parametric shortest path and minimum-balance algorithms. *Networks*, 21:205–221, 1991.

Index

- Ahuja, R., 91, 121
 Aspvall, B., 60, 121
 Bard, J., 30, 121, 125
 Ben-Ameur, W., 23, 121
 Bialas, W., 30, 121
 Bouhtou, M., 26, 42, 53, 57, 60, 73,
 119, 121, 123, 125, 131
 Bracken, J., 30, 121
 Braess, D., 26, 122
 Brothorne, L., 21, 26, 86, 122
 Burkard, R.E., 71, 122
 Calamai, P.H., 30, 125
 Candler, W., 30, 122
 Chvátal, V., 31, 86, 108, 122
 Côté, J.P., 26, 122
 Cole, R., 26, 122
 Dal, R., van, 71, 122
 Dantzig, G., 53, 122
 Deineko, V.G., 71, 122
 Dempe, S., 30, 122
 Dewez, S., 88, 122
 Diallo, M., 26, 121
 Didi, M., 89, 90, 120, 122, 132
 Dodis, Y., 26, 122
 Falk, J., 30, 121
 Fortuny-Amat, J., 30, 123
 Fortz, B., 53, 123
 Fulkerson, R., 53, 122
 Garey, M.R., 53, 54, 80, 82, 123
 Gauvin, J., 30, 125
 Geffard, J., 23, 121
 Gourdin, E., 23, 121
 Grigoriev, A., 53, 57, 73, 123
 Hansen, P., 30, 123
 Håstad, J., 82, 123
 Hoesel, C.P.M. van, 42, 53, 57, 60, 73,
 119, 121, 123, 125, 131
 Ishizuka, Y., 30, 125
 Jahn, O., 26, 123
 Jaumard, B., 30, 123
 Jeroslow, R.G., 30, 123
 Johnson, D., 53, 54, 80, 82, 123
 Johnson, S., 53, 122
 Júdice, J., 30, 125
 Karmarkar, N., 60, 123
 Karp, R.M., 64, 123
 Karwan, M., 30, 121
 Khachian, L.G., 60, 123
 Kolstad, 30, 123
 Kraaij, A.F., van der, 42, 53, 57, 60,
 73, 119, 121, 123, 125, 131
 Labbé, M., 21, 26, 30, 35-37, 39, 53,
 60, 66-68, 84-86, 88, 94, 96,
 122-124
 Lasdon, L., 30, 123
 Lawler, E.L., 71, 124
 Lenstra, J.K., 71, 124
 Liao, B., 23, 121
 Liu, L.N., 26, 124
 Luo, Z.Q., 30, 124
 Lutton, J-L., 42, 121
 MacKie-Mason, J., 26, 124
 Magnanti, T., 91, 121
 Mannino, C., 53, 60, 125
 Marcotte, P., 20, 21, 26, 29, 30, 35-37,
 39, 53, 54, 57, 60, 66-68, 73,
 82, 84-86, 88-90, 94, 96, 119,
 120, 122-124, 131, 132
 McCarl, B., 30, 123
 McDonald, J.F., 26, 124
 McGill, J., 30, 121
 Michel, N., 23, 121

- Motwani, R., 82, 124
 Möhring, R., 26, 123
 Oriolo, G., 53, 60, 125
 Orlin, J., 64, 91, 121, 123, 125
 Pang, J.S., 30, 124
 Papadimitriou, C.H., 53, 124
 Raghavan, P., 82, 124
 Ralph, D., 30, 124
 Rinnooy-Kan, A.H.G., 71, 124
 Roch, S., 20, 37, 53, 54, 57, 66, 73, 82, 119, 124, 131
 Roughgarden, T., 26, 53, 122, 124
 Savard, G., 20, 21, 26, 29, 30, 35-37, 39, 53, 54, 57, 60, 66-68, 73, 82, 84-86, 88-90, 94, 96, 119, 120, 122-125, 131, 132
 Schulz, A., 26, 123
 Semet, F., 53, 124
 Shimizu, K., 30, 125
 Shmoys, D.B., 71, 124
 Stackelberg, H., 18, 29, 125
 Steiglitz, K., 53, 124
 Stevens, W.R., 21, 125
 Stone, R.E., 60, 121
 Tardos, É., 26, 124
 Tarjan, R.E., 64, 125
 Thorup, M., 53, 123
 Tirole, J., 29, 125
 Townsley, R., 30, 122
 Uetz, M., 53, 57, 73, 123
 Varian, H., 26, 29, 124, 125
 Veen, J.A.A., van der, 71, 122
 Verhoef, E.T., 26, 125
 Vicente, L.N., 29, 30, 125
 Woeginger, G.J., 71, 122
 Wynter, L., 26, 121
 Young, N.E., 64, 125
- All-service network tariffication problem, 25
 ASNTP, *see* All-service network tariffication problem
- Bilevel programming, 29
 Bilinear-linear, 30
 Linear-linear, 30
- Linear-quadratic, 30
 Lower level, 30
 Quadratic, 30
 Upper level, 30
- Bilinear, 30
 Bounded arcs linear tariffication problem, 60
- CPLEX, 95, 112, 122
- EU Directive, 17, 122
- Fixed path linear tariffication problem, 66
 Follower, 29
 Follower's problem, 30
- Leader, 18, 29
 Leader's problem, 30
 LEDA, 112, 124
 Lower bound, 36
 Cost of path, 36
 Revenue, 92
- Multi-commodity fixed path tariffication problem, 66
- Network tariffication problem, 18, 29
 NTP, *see* Network tariffication problem
- Parametric tariffication problem, 64
 Path dominance, 46
 Pricing strategy, 23, 32
 Fixed charge, 32
 Inclusive minutes, 32
 Linear, 32
 Parametric, 33
 Piecewise linear, 33
 Staircase, 33
 Uniform, 33
- PTP, *see* Parametric tariffication problem
- Relevant path, 48
 River tariffication problem, 24, 47
 Bounded number of commodities, 71

1. The first of these is the fact that the
 2. the second is the fact that the
 3. the third is the fact that the
 4. the fourth is the fact that the
 5. the fifth is the fact that the
 6. the sixth is the fact that the
 7. the seventh is the fact that the
 8. the eighth is the fact that the
 9. the ninth is the fact that the
 10. the tenth is the fact that the

1. The first of these is the fact that the
 2. the second is the fact that the
 3. the third is the fact that the
 4. the fourth is the fact that the
 5. the fifth is the fact that the
 6. the sixth is the fact that the
 7. the seventh is the fact that the
 8. the eighth is the fact that the
 9. the ninth is the fact that the
 10. the tenth is the fact that the

Samenvatting

In hoofdstuk 1 en 2 van dit proefschrift introduceren we een prijsbepalingsprobleem, het zogenaamde *network tariffication problem*. Voor dit prijsbepalingsprobleem is een netwerk gegeven met daarin een aantal kanten waarvoor de prijs al bekend is en een aantal kanten waarvoor dit niet het geval is. De prijs op deze laatstgenoemde kanten, de tariefkanten, moet nog nader bepaald worden. Klanten in het netwerk willen hun data van één punt in het netwerk naar een ander punt krijgen, tegen de goedkoopst mogelijke prijs. De prijzen op de tariefkanten worden door de aanbieder van capaciteit in het netwerk echter op de winstmaximaliserende prijs gezet.

Een hermodellering van het netwerk wordt besproken, welke zich beperkt tot de combinatorische structuur van het probleem: het *shortest path graph model*. Naast het geven van meer inzicht in de structuur van het netwerk, wordt deze hermodellering in latere hoofdstukken gebruikt om het prijsbepalingsprobleem optimaal op te lossen. France Télécom deed een patentaanvraag voor deze hermodellering, welke door het *Institut National de la Propriété Industrielle* in Parijs, Frankrijk, werd toegekend op 15 augustus 2003 (zie [9]) en door de *World Intellectual Property Organization* op 21 augustus 2003 (zie [8]).

De complexiteit van het prijsbepalingsprobleem wordt geanalyseerd in hoofdstuk 3. Aan de hand van het bewijs van Roch et al. [53] wordt aangetoond dat het probleem \mathcal{NP} -moeilijk is. We bewijzen dat twee speciale gevallen van het probleem \mathcal{NP} -moeilijk blijven en bespreken verschillende nieuwe polynomiaal oplosbare problemen. Een van de speciale gevallen die polynomiaal oplosbaar zijn, is het prijsbepalingsprobleem met een lineaire prijsstrategie op de tariefkanten waarvoor het aantal tariefkanten in het netwerk begrensd is. Een ander speciaal geval is het prijsbepalingsprobleem met een parametrische prijsstrategie op de tariefkanten.

In hoofdstuk 4, een hoofdstuk gewijd aan approximatiere resultaten, wordt besproken dat uniforme prijsbepaling een $|T|$ -benaderingsalgoritme levert voor het speciale geval van het prijsbepalingsprobleem waar de tariefkanten zich in een snede bevinden (het aantal tariefkanten is gelijk aan $|T|$). Wanneer de kosten op de vaste kosten kanten in het netwerk polynomiaal begrensd zijn in $|T|$, kunnen we het algoritme verbeteren tot een $\mathcal{O}(\log(|T|))$ -benaderingsalgoritme. In hoofdstuk 4 wordt verder bewezen dat wanneer alle klanten in het netwerk bediend moeten worden (het aantal klanten is gelijk aan $|K|$), het prijsbepalingsprobleem niet benaderbaar is met $\mathcal{O}(|T|^{1-\epsilon})$ of met $\mathcal{O}(|K|^{1/2-\epsilon})$, tenzij $\mathcal{ZPP} = \mathcal{NP}$.

De complexiteit van het prijsbepalingsprobleem met een niet-lineaire prijsstrategie en meerdere tariefkanten, waarbij het aantal tariefkanten begrensd is, is op dit moment onbekend. Het is verder op dit moment nog onbekend of er betere benaderingsalgoritmen zijn voor het speciale geval van het prijsbepalingsprobleem waar de tariefkanten zich in een snede bevinden en of we de resultaten voor dit speciale geval kunnen overbrengen naar het prijsbepalingsprobleem met meerdere klanten. Verder is het op dit moment nog onbekend of het prijsbepalingsprobleem niet te benaderen is wanneer niet alle klanten bediend hoeven te worden.

In hoofdstuk 5 bespreken we exacte oplossingsmethoden voor het prijsbepalingsprobleem met een lineaire prijsstrategie op de tariefkanten. Hoewel, zoals in hoofdstuk 2 besproken, de bovengrens op het aantal ongedomineerde paden van een klant gehaald kan worden voor kunstmatige instanties, laten de rekenresultaten de kracht van de hermodellering van het netwerk zien op de geteste instanties: over het algemeen wordt maar een klein aantal ongedomineerde paden gegenereerd aan de hand van de hermodellering. In hoofdstuk 6 worden de exacte oplossingsmethoden van hoofdstuk 5 gegeneraliseerd naar alle prijsstrategieën. Het gevolg van het gebruik van elke prijsstrategie op de opbrengst voor de aanbieder van capaciteit in het netwerk en op de kosten voor de klanten wordt onderzocht, aan de hand van een aantal instanties van France Télécom.

Hoofdstuk 5 en hoofdstuk 6 laten zien dat de hermodellering van het netwerk goede resultaten levert en dat formuleringen, gebaseerd op deze hermodellering erg efficiënt zijn. In verder onderzoek zou deze hermodellering kunnen worden toegepast op andere formuleringen van het prijsbepalingsprobleem. In recentelijk onderzoek van Didi et al. [26] wordt een aantal nieuwe formuleringen hiervoor besproken. Deze formuleringen werden niet geïmplementeerd en getest, maar het zou interessant zijn om ze te vergelijken met de huidige oplossingsmethoden. Toekomstig onderzoek zou zich kunnen richten op het prijsbepalingsprobleem met capaciteiten op de kanten in het netwerk of op de optimalisatie van de grootte van de lineaire stukken en van het aantal lineaire stukken in een prijsstrategie.

About the Author

Anton Frederik van der Kraaij was born in the Eternal Love Winning Africa (ELWA) hospital in Monrovia (Liberia) on September 2nd, 1977. Until 1991, he attended French international schools in Burkina Faso, Mali and Senegal. In 1991, he continued his high school at the Almere College in Kampen (The Netherlands). After receiving his Atheneum Diploma in 1995, he decided to study Econometrics at Maastricht University in Maastricht (The Netherlands). His Master's thesis was written during an internship at France Télécom Research & Development in Paris (France) and completed in 2000. In the same year, the cooperation between Maastricht University and France Télécom Research & Development continued when he started as a PhD candidate for his research on the subject of this thesis.

